Sustainable Computing: Informatics and Systems 33 (2022) 100622

Contents lists available at ScienceDirect

Sustainable Computing: Informatics and Systems

journal homepage: www.elsevier.com/locate/suscom

Internet of Things for sustaining a smart and secure healthcare system

Prabh Deep Singh [a], Gaurav Dhiman [b], Rohit Sharma [c,*]

[a] Department of Computer Science and Engineering, Graphic Era (Deemed to be University), Dehradun, Uttarakhand, India
[b] Department of Computer Science, Government Bikram College of Commerce, Patiala, India
[c] SRM Institute of Science and Technology, Ghaziabad, India

ARTICLE INFO

Keywords:
Thyroid
Artificial intelligence
Fog computing
Cloud computing
Internet of Things

ABSTRACT

The thyroid is a key endocrine gland in the human body that regulates several bodily processes, including protein synthesis, energy consumption, and the body's reaction to other hormones. Segmentation and volume regeneration of the thyroid is particularly important for identifying thyroid-related diseases since the majority of these problems result in a change in the thyroid's shape and scale over time. There is an urgent need for research on the disease's origins and spread. The Internet of Things, cloud computing, and artificial intelligence all provide real-time processing for a variety of applications in the healthcare sector. In healthcare and biomedicine applications, machine learning algorithms are increasingly being utilized to make critical choices. Thyroid patients urgently need a robust and latency-sensitive Quality of Service framework. This paper aims to integrate fog computing and artificial intelligence with smart health to provide a dependable platform for thyroid infection early detection. To identify thyroid patients, a novel ensemble-based classifier is proposed. The thyroid dataset is obtained from the UCI library and the simulation is carried out utilizing Python programming. To increase the framework's security, encryption and decryption methods are suggested. The suggested framework's performance is assessed in terms of latency, network use, RAM utilization, and energy consumption. On the other side, the suggested classifier's accuracy, precision, specificity, sensitivity and F1 score are all assessed. The result demonstrates that the suggested framework and classifier perform consistently better than conventional frameworks and classifiers.

1. Introduction

Thyroid disease is a chronic condition that affects almost 200 million people globally and is more prevalent in women than men. Thyroid disorder in women is difficult to diagnose and treat due to a variety of causes. Many women with thyroid disease are misdiagnosed or continue to have symptoms even after being treated. Numerous causes complicate the diagnosis and treatment of thyroid disease. The most often used blood test for thyroid disorder (thyroid-stimulating hormone [TSH]) can yield false results. More precisely, while combining all three major thyroid function tests (TSH, free T3, and free T4) provides the most comprehensive assessment of thyroid function, the TSH test is often used alone. As a result, critical health diagnoses (e.g., hyperthyroidism, hypothyroidism) are sometimes overlooked. Additionally, since thyroid dysfunction mimics the effects of other diseases (e.g., stress, menopause), thyroid disease is often misdiagnosed and untreated.

Today's age of ICT IoT, cloud computing, fog computing, and artificial intelligence has accomplished numerous marvels that human beings could not have imagined. The primary goal of smart healthcare technology is to offer an adequate and efficient service that meets the needs of all healthcare sectors. The rapid growth and development of the IoT have made it a critical component of the healthcare industry [1–4]. In the actual world, a highly precise health monitoring mechanism and associated processing are critical in an IoT context. Regulation and treatment of thyroid major instances may be accomplished with the help of IoT technology without imposing severe restrictions on people and companies. With the recent success of AI applications in the healthcare industry, it may be critical for avoiding unnecessary deletions and optimizing the efficacy and efficiency of research using huge datasets. It may be extensively utilized with thyroid; furthermore, we aim to find the best potential solutions to the thyroid issues that have posed the most important barriers to health care systems via the use of new technology. Doctors may remotely access patient information and health status and take necessary action to save the patient's life. It is advantageous and essential to bring the cloud closer to the patient/doctor for real-time health applications [5–8].

* Corresponding author.
E-mail addresses: ssingh.prabhdeep@gmail.com (P.D. Singh), gdhiman0001@gmail.com (G. Dhiman), rohitr@srmist.edu.in (R. Sharma).

https://doi.org/10.1016/j.suscom.2021.100622
Received 25 May 2021; Received in revised form 17 September 2021; Accepted 30 October 2021
Available online 17 November 2021

2. Related work

Medical diagnosis is the method of deciding which illness or disorder accounts for a patient's symptoms and signs [9,10]. Typically, the details necessary for evaluation are gleaned from the patient's experience and clinical evaluation. Diagnosis is the most important step toward treating a patient. Diagnosis is often challenging since many illnesses and disorders have the same main symptoms. For instance, fever and headache are common symptoms associated with any illness, which makes determining the underlying cause challenging for healthcare professionals. Diagnostics should be seen statistically as a classification issue in which a piece of knowledge assists us in determining the true cause and identifying the right illness or disorder. Medical evaluation is the most important aspect of health care. However, the identification is challenging due to many illnesses with the same signs.

Fog computing is one of the most rigorously developed solutions for applications that need low latency, reduced network congestion, and rapid response [11,12]. It addresses issues such as edge location, high latency, location knowledge, durability, and data transmission to the appropriate processing site. In today's smart health care environment, physicians and patients benefit from the use of new technologies such as IoT, cloud computing, and fog computing since these technologies are applied to many medical sectors such as hospital administration. Management of patient records, real-time monitoring of patients, and so on. The majority of fog cloud healthcare designs do not include AI methods to get optimal outcomes. The pace of adoption of smart healthcare is contingent upon the degree of security offered by the health apps [13,14].

Artificial Intelligence (AI) applications in healthcare have been investigated, and an increasing number of businesses are emerging in this domain [15–19]. Though the growth of AI in healthcare has been slow and consistent, the lack of end-to-end systems has not prevented the AI group from assisting doctors and automating multiple activities involved in the pipeline, from diagnosis to treatment. Many of the recent advances in applying AI to healthcare can be since larger databases are being rendered freely accessible by healthcare agencies and that the computer capacity available to mankind is growing by orders of magnitude per year.

Health forecasting is a novel field of forecasting and an extremely useful method for forecasting potential health events or conditions, such as demand for health insurance and healthcare requirements [20]. It promotes preventive medicine and health care intervention plans by advising health care professionals in advance to take effective risk-reduction and demand-management measures. Health forecasting necessitates the use of accurate data, knowledge, and computational methods to forecast particular health problems or circumstances. Predictive modeling analyses current and historical data obtained by medical institutions using computational techniques, data processing, and game theory. These data contribute to the improvement of medical treatment and the achievement of positive clinical outcomes.

Predictive analytics is the method of extracting information from past data to render possible forecasts [21]. It can help the right choices to be made in health care, allowing for more personalized care. Predictive analytics in healthcare can assist in detecting early symptoms of patient decline in the intensive care unit and general ward, identifying at-risk patients at home to reduce hospital readmissions, and preventing avoidable equipment downtime [22]. It helps to notify physicians and patients about the risk of occurrences and consequences in advance, assisting them in preventing as well as curing health problems. Machine learning is a well-researched field with a strong track record of progress in a variety of industries. Healthcare should use this prior experience to accelerate the adoption of predictive analytics for optimizing patient care, chronic condition management, hospital leadership, and supply chain efficiency. The opportunity for healthcare providers at the moment is to identify what "predictive analytics" represents to them and how it can be applied more efficiently to optimize outcomes.

Nonetheless, deep learning is more suited to some systems than others [23–27]. Algorithms may immediately support disciplines with reproducible or systematic procedures. Additionally, those working in fields of massive picture datasets, such as radiology, cardiology, or pathology, are excellent candidates. Machine learning algorithms may be programmed to analyze videos, detect anomalies, and highlight areas that need focus, thus increasing the precision of both of these methods. Machine learning will eventually benefit the family physician or internist at the bedside. Machine learning may include an analytical assessment of a situation to increase performance, reliability, and accuracy [28–31].

The following challenges are identified while designing smart health applications.

1. Authentications of stakeholders are required while accessing the data.

2. Authentication of machines or sensors is required for those who are transferring the data.

3. The security of data stored in a cloud or fog database is an essential requirement.

This paper's main contribution is summarized as follows.

- Provide a more accurate solution that allows the estimation of thyroid at a very early stage with ultra-low latency.
- The artificial intelligence is used to evaluate the patients' past medical details and forecast thyroid cases, which can aid in lowering costs and distinctively enhance the standard of treatment at hospitals.
- Address the security requirements for the proposed framework and provide a solution using encryption and decryption techniques. The performance of the classifier is evaluated in terms of classifier's accuracy, precision, specificity, sensitivity and F1 score.
- Describing the result in a specific way through repeatedly conducting experiments measure QoS parameters like latency, energy consumption, network cost, and security, which can guarantee the efficiency of the proposed framework.

The remaining paper is organized in the following way. In Section 2, related work is presented. Section 3 proposed a Quality of service aware smart health Framework for the thyroid. Experimental setup and Performance Evaluation are presented in Section 4. The security aspects of the proposed framework is discussed at Section 5. Section 6 closes the paper with a concluding remark and proposes a direction for future development.

3. Proposed framework

Fig. 1 shows the proposed framework which is divided into three sub systems.

3.1. User subsystem

The proposed User Subsystem offers an interface via which all stakeholders in the smart healthcare system may interact and obtain/provide the necessary information. Doctors may do background checks on patients, monitor them remotely, and update their records. Additionally, they may keep the patient's current health record. Experts who are specialists in their fields may examine their data, produce reports, and keep current on viral illnesses and newly found drugs. Patients may engage with a variety of wearable Internet of Things devices and wireless health monitoring equipment such as a glucose meter, cardiac monitor, and blood pressure monitor. These integrated gadgets provide real-time monitoring of patients. Numerous gadgets are linked to the internet and constantly transmit data for storage and analysis. This interface is used to save data in the repository.

Data collected from different devices is automatically sent to powered-on devices, agents. Additionally, data is sent to the Fog

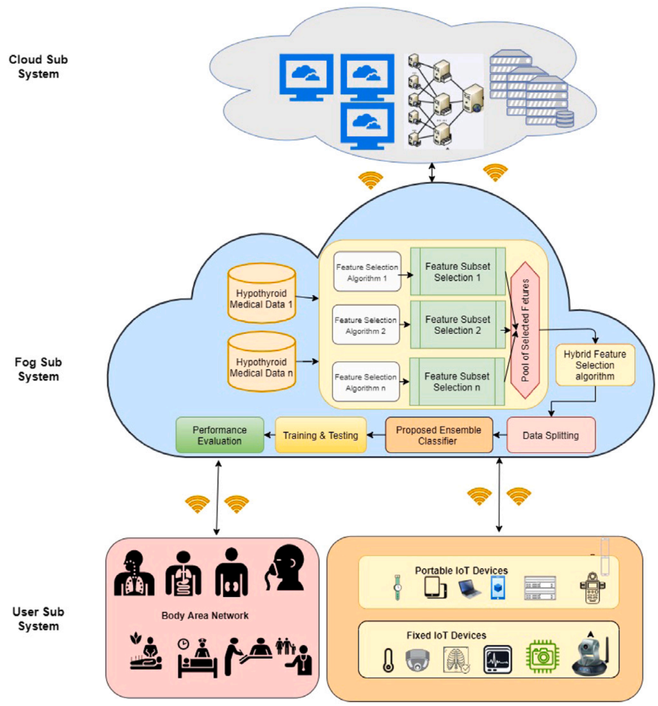

Fig. 1. Proposed system overview.

Subsystem. Here, data is mined and filtered using a variety of artificial intelligence techniques. The Fog Subsystem stores a copy of sensitive data and sends the full data to the cloud subsystem; the fog repository stores the address of the remaining data. There are many practical applications of ranking in machine learning, including the ability to discover an instance's particular label preference through a complete ranking. The specialist doctor shares their experience by accessing patient queries and updates the database on the latest diseases explored in their field. Various experts contribute their research in this repository, and patients get the benefit.

3.2. Fog subsystem

The Fog Subsystem plays an essential role in the real-time processing of patient data. This Fog Subsystem is capable enough to collect data from the IoT Devices, process it, store it, and generate alerts accordingly. As fog nodes are located closer to the patient, data transmission takes occurs quickly, resulting in reduced network latency. Fog nodes are very compact and intelligent devices with extremely limited processing (CPU, memory, and storage) capabilities. The Artificial Intelligence Unit determines whether or not the user is infected with thyroid. Following that, if it is determined that the patient has a thyroid infection, his/her information is updated in the medical database. The primary aim of this unit is to collect the data and process it using different artificial intelligence methods. The data are transferred in several stages. Each stage of the AI Unit plays an amazing significance in the success of the procedure. One step output becomes the next step input. Each stage's success relies on the quality of the preceding step. The fog repository is installed in the fog device and is connected to the cloud repository via USB. It is used to hold patient information. The Fog Subsystem stores a copy of sensitive data and sends the full data to the cloud subsystem; the fog repository stores the address of the remaining data. The Fog Subsystem is critical for real-time patient data processing. Because fog nodes are located closer to the patient, data transmission takes occurs quickly, resulting in reduced network latency. Fog nodes are distributed fog computing entities that make fog services possible to deploy. They are comprised of at least one or more physical devices with computing and sensing capabilities. The Fog Subsystem is equipped with real-time processing and storing capabilities. Additionally, it alleviates cloud network congestion.

Table 1
Data type for different symptoms of hypothyroid patients.

S. No.	Parameter	Data type
1	Age	Integer
2	Sex	Boolean
3	Thyroxine	Boolean
4	Antithyroid medication	Boolean
5	Thyroid surgery	Boolean
6	Pregnant	Boolean
7	Sick	Boolean
8	Tumor	Boolean
9	Lithium	Boolean
10	Goitre	Boolean
11	TSH	Float
12	FTI measured	Boolean
13	TBG measured	Boolean

3.2.1. Data description

Datasets play a critical role in machine learning. Data collection is one of the most challenging practices, perhaps more so when it comes to therapeutic domains such as early detection. The Table 1 summarizes the data types used to describe the various symptoms experienced by hypothyroid patients. Hypothyroid infected and hypothyroid uninfected cases are included in the dataset. There are 3163 cases in the dataset, but only 151 are hypothyroid infected. Since the infected class accounts for only 4.77% of the total data.

3.2.2. Feature selection

Feature selection is a widely used strategy in data preprocessing for machine learning that distinguishes critical features and eliminates obsolete, redundant, or noise features to low the dimensionality of the feature room. It enhances the performance, precision, and comprehensibility of models constructed using machine learning algorithms.

Feature Selection Procedure A standard protocol for selecting features consists of four fundamental steps: (1) Generation of subsets; (2) assessment of subsets; (3) selection of a stopping criterion; and (4) outcome validity. The method starts with feature subset creation, which utilizes a specific search technique to generate nominee feature subsets. Following that, each applicant subset is assessed using a specific assessment criterion and contrasted to the previous best. If it is superior, it supersedes the previous best. The process of creating and evaluating subsets is replicated until a defined stopping criterion is met. Finally, the best function subset chosen is tested using previous information or test results. Two critical concepts in the analysis of feature selection are search strategy and assessment criteria.

Generation of subsets Subset generation starts with a starting point for the quest, which may be an empty set, the complete set, or a randomly created subset. It will look for function subsets in a variety of ways, including forwarding, backward, and random. Forward search adds functionality one by one, while backward search removes the least relevant function depending on the evaluation criteria. To prevent getting stuck in a nearby maximum, random search introduces or deletes functionality at random. There are several search methods for determining if a function subset is optimal or suboptimal. As we note, if the total number of features in the function set is N, the total number of nominee subsets is 2N. A search technique that is comprehensive scans all 2N function subsets to locate the optimal one. In terms of data dimension, its complexity is exponential. As extended to large datasets, determining the best function subset is often intractable. As a result, numerous heuristic search techniques have been created to address this problem.

Assessment of Subsets After generating function subsets, they are tested against a predefined criterion to determine their goodness. In general, the goodness of function subsets refers to their capacity to discriminate between distinct groups. Feature selection algorithms may be roughly classified into three groups based on their reliance on inductive learning algorithms: wrapper [32], filter [33], and hybrid.

Hybrid Method To boost classification efficiency and accelerate feature collection, one may create hybrid models that leverage filters and wrappers by measuring feature subsets using both independent parameters and learning algorithms [34]. Filters may have an intelligent guideline for wrappers, such as a reduced search space, a decent starting point, or a smarter/shorter search route, both of which assist wrappers in scaling to larger size issues. Typically, a hybrid approach employs an independent calculation to determine the best subsets for a given cardinality and a learning algorithm to determine the final best subset of the best subsets through various cardinalities. It usually begins with an empty subset and iterates to find the better subsets in increasing cardinality order.

3.2.3. Selection of a stopping criterion

A feature selection phase may be halted for either of the following reasons:

- Whether or not the quest has been completed.
- Whether or not to use a predefined scale for function subsets.
- Whether or not a defined number of iterations were performed.
- If an ideal or reasonably strong feature subset has been achieved in terms of the evaluation function.
- If altering function subsets would not result in a better subset.

Outcome Validity In certain cases, the related characteristics are identified in advance. Then, using this previous information, we will verify the function selection findings. However, in the majority of real-world implementations, we have no idea which characteristics are important. We must use the classification output on test data as a proxy for the goodness of the function subsets we have chosen.

Data splitting For each experiment, we split the entire dataset into a training set of 68% and a test set of 32%. We resampled, set hyperparameters, and trained the model using the training set, and then evaluated how well the learned model worked. When dividing the data, we defined a random seed to ensure that the same data was divided each time the program ran [35].

3.2.4. Proposed ensemble model

Ensemble methods are one of the most popular and effective multi-classifier approaches which consists in combining a set of classifiers of the same type to get a single more efficient model [36]. Nowadays, many methods are automatically capable of generating sets of classifiers such as Bagging, Boosting, Random Subspaces, random Forest, Extra-trees. The random forests method is the most popular among the ensemble methods. This method is bagging improved to the level of hyperparameters. It is based on the combination of the elementary classifiers of the decision trees type. Individually, these classifiers are not effective, but they have interesting properties to operate within an ensemble as they are particularly unstable. The specificity of the trees used in random forests is that their induction is disturbed by a random factor, and the purpose is to generate diversity in the ensemble. It is based on these two elements: the use of decision trees as elementary classifiers and the introduction of randomness in their induction that the formalism of random forests was introduced. The existing methods in the state of the art try to deal with randomization. A good ensemble method should ensure the optimal level of randomization which minimizes the error rate and the variance. However, existing methods still have this limit. Our first contribution in this proposal is the introduction of a new method to generate a set of classifiers. This method combines Bootstrap Sampling, Random Subspaces, and random forests to generate a more efficient set of trees than each method individually. This method has been tested and has proven its effectiveness vis-a-vis several methods in the literature. In this paper, we deal also with the problem of trees aggregation and ensemble selection in the tree-based ensemble methods. Classical random forests use majority voting to aggregate the decision of each classifier. This technique is not optimal since it gives the same weight to the decision of each tree even they do have not the same performances. In our second contribution, we propose a weighted voting mechanism to random forests which give better results than the classical majority voting. The third contribution is a tree selection method in a forest to keep only the best trees. This technique belongs to the family of ensemble selection or pruning methods. All existing dynamic pruning methods in the state of the art use KNN (K-nearest neighbor) as a neighborhood heuristic. In our dynamic pruning method called Out of Bag-Based Ensemble Pruning, we propose the use of a different neighborhood heuristic that uses the path similarity between the test instance and the Out of Bag of this tree. This pruning method is used to select, for each test instance, a subset of different trees of the forest. The class of that instance is assigned through a majority vote between the results returned by the trees of the selected subset.

3.2.5. Training and testing phase

The Test dataset contains the data that will be assessed. It is used only once during a training run. The reference database is used to evaluate competing models. Sometimes, the validation collection is used as a test set, although this is not a good strategy since it will result in a mess. The test range was purposefully selected for precision. The model incorporates widely available data gathered by a variety of groups on which the model would be tested. After adjusting the hyperparameters on each predictive model, we passed the reequipped training set as training data on each model. Thus, the novel ensemble-based algorithm learned distinct trends in the resampled training results—the test array we had previously isolated to test the model performance when breaking the entire dataset.

3.3. Cloud subsystem

Three kinds of information are kept in this subsystem. Initially, the patient is kept with basic information such as name, id, age, gender. Secondly, it stores patients' behaviors and professional habits, such as smoking, drinking, computer work, dancing, gym, etc. The third kind of information is information in real-time such as the pulsation rate, sleep, walking time, and heartbeat. For each patient, an automated patient Id is created. Each patient generates and stores a significant quantity of data with their ID. High-end servers in the cloud subsystem are installed. This database uses AI algorithms, and its findings are shared with specialists and physicians. Doctors may also monitor their patients in real-time. This Cloud Subsystem contains 3 modules i.e. Data repository, Policy unit, and device manager

3.3.1. Data repository

The framework uses two data repositories, that is, the cloud repository and the fog repository. The fog repository provides the data to the AI device, and it is synchronized with the cloud repository. All data collected from different IoT devices are initially saved in the fog repository. Further data analyses are carried out. The whole data is sent to the cloud repository. Only these data stay in the fog repository, chosen by AI algorithms at the first stage.

3.3.2. Policy unit

The policy unit determines the patient data storage method that data may be accessed, stored, or changed. Patients must regulate with the cloud repository and be authorized to store/read/modify their data and other information. The initial reference to its first block is provided by the cloud repository. Patient information cannot be saved in the cloud repository directly. The data must be extracted and encrypted using the shared key.

3.3.3. Device manager

The Device Manager's primary responsibility is to collect the data from various IoT-based medical devices and send them to the fog module. These IoT-based medical devices are battery-operated, whereas the device manager is a powered-up device.

4. Security aspects of proposed framework

While security criteria are a critical component of any cloud/fog framework, there is a lack of techniques provided for designing a secure software system. In the proposed framework encryption and decryption is utilized to provide the security of patient data while it is stored in servers or traveling from one subsystem to other subsystem. Following algorithms are used for encryption and decryption.

Encryption

Divide M_{ENCP} into $M_{ENCP}[1]$ $M_{ENCP}[2]$ ... $M_{ENCP}[MAX]$
$L<-EK(0n)$
$R<-EK(N+L)$
for $i<-1$ to MAX
do $B[i]=L+R$
for $i<-1$ to $MAX-1$
do $M_{DECP}[i]<-EK(M_{ENCP}[i])+B[i]+B[i]$
$X[MAX]$ =lgt($M_{ENCP}[MAX])+L+B[MAX]$
$Y[MAX]=EK(X[MAX])$
$M_{DECP}=M_{DECP}[1]M_{DECP}[2]....M_{DECP}[MAX]$
$checksum=M_{ENCP}[1]+M_{ENCP}[2]..+M_{ENCP}[MAX-1]+M_{DECP}[MAX]0*+A[MAX]$
$T=EK(checksum+B[MAX])$
$ReturnM_{DECP}=M_{DECP}\parallel T$

Decryption

Divide M_{ENCP} into $M_{ENCP}[1]$
$M_{ENCP}[2]...M_{ENCP}$ [MAX]T
$L<-E_K(0^n)$
R<-$E_K(N+L)$
for i<-1 to MAX
do B[i]=L+R
for i<- 1 to MAX-1
$M_{ENCP}[i]=E_K(M_{DECP}[i]+B[i]+B[i])$
$X[MAX]=lgt(M_{DECP}[MAX])+L+B[MAX]$
$Y[MAX]=E_K(X[MAX])$
$M_{ENCP}=M_{ENCP}[1]M_{ENCP}[2]....M_{ENCP}[MAX]$
$checksum=M_{ENCP}P[1]+M_{ENCP}[2]..+M_{ENCP}[MAX-1]+M_{DECP}[MAX]0*+A[MAX]T=EK(checksum+B[MAX])$
$ReturnM_{DECP}$

The message from the source node denoted by M_{ENCP}, requires encryption. The encryption key is symbolized as K where the complete source message is represented as $M_{ENCP}=M_{ENCP}[1], M_{ENCP}[2], M_{ENCP}[3], \dots M_{ENCP}[MAX]$ here MAX =Maximum1, $[M_{ENCP}/n]$ and $M_{ENCP}[1]=M_{ENCP}[2]=\cdots.M_{ENCP}[MAX-1]=n$. Here, the value of n is also needs encryption which is selected either by the source or destination and it is non-repeating in nature. On the other hand, the decrypted this message is shown as $M_{DECP}=M_{DECP}[1]M_{DECP}[2]$ $M_{DECP}[3]...M_{DECP}[MAX]$ in the form of cipher text. The checksum is given as $M_{ENCP}[1]+M_{ENCP}[2]+\cdots M_{ENCP}[MAX-1]+M_{DECP}[MAX]0*+A[MAX]$ with offset $B[1]=L+R$. E_K is applied to the 0^n, which is a fixed string to define the string L.

5. Experimental setup and performance evaluation

The experiments are performed in two folds. Firstly the entire framework is simulated to measure the delay, Network Usage, RAM usages, and, Energy Consumption. The proposed classifier is evaluated in the second fold in terms of classifier's accuracy, precision, specificity, sensitivity and F1 score.

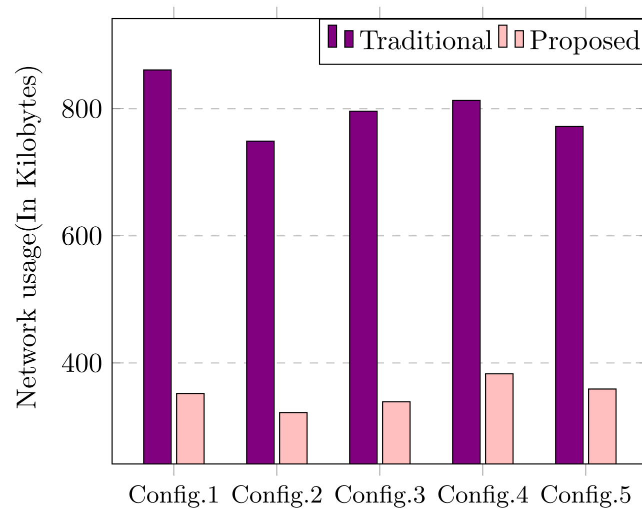

Fig. 2. Comparison of network usage.

5.1. Performance evaluation of proposed framework

iFogSim Tookit [37] is chosen as a test-bed for performance evaluation of the proposed framework. To execute the simulation, the CloudSim libraries are included. The simulation takes place in two scenarios. In the first scenario, the IoT subsystem's data is transmitted by the intermediate fog subsystem to the cloud subsystem. If the user needs to access the data, the request is sent via the fog subsystem to the cloud subsystem, and the data is retrieved. In the second scenario, the data generated is transmitted directly to the cloud subsystem without the fog subsystem being involved. On the other side, the request is immediately transferred to the cloud subsystem and retrieved when the user wants to access the data.

5.1.1. Delay

Measuring the Round Trip Time (RTT) delay for data transmission from the IoT sub system to the accessing node, the delay time is measured.RTT delay is measured both for first scenario in the presence of fog sub system and for the second scenario in the absence of sub system, are shown graphically in Fig. 5.

5.1.2. Network Usage

Fig. 2 describes the network use of the proposed framework. A large rise in the amount of mobile devices raises the network demand where only cloud sub system is utilized. As Fig. 2 indicates, network use declined dramatically when fog sub system is included.

5.1.3. RAM usage

RAM uses are determined by calculating the heap assignments during simulations of different configurations. As seen in Fig. 3, where only the cloud subsystem is regarded, the heap allocation is increased. The proposed framework always consume less amount of RAM as compared with the traditional frameworks

5.1.4. Energy consumption

Fig. 4 explains the energy expended in the simulation by various classes of IoT units. As Fig. 4 indicates, when the traditional framework has been used, the energy consumption of various IoT devices is more as the energy consumption of IoT devices is decreased drastically when the proposed framework is used.

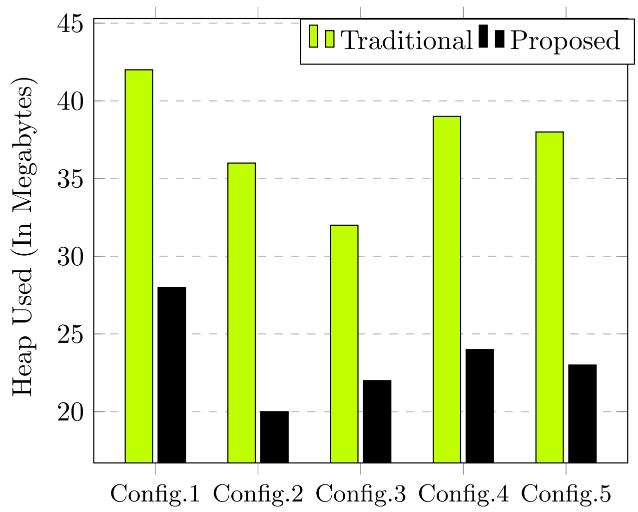

Fig. 3. RAM usage.

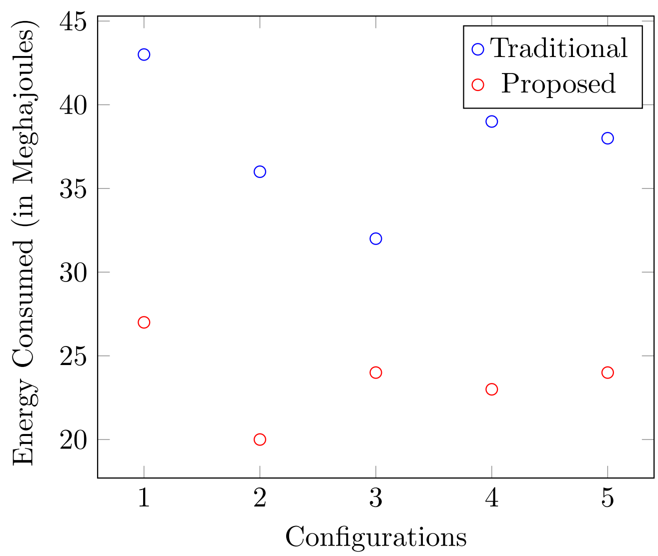

Fig. 4. Energy consumption of devices.

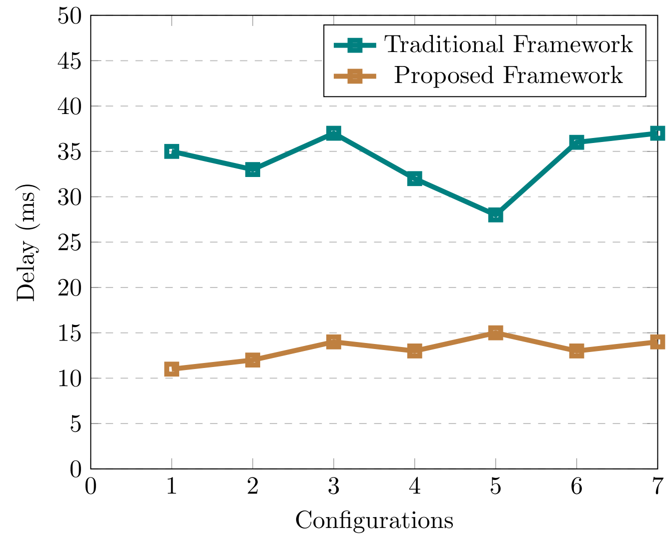

Fig. 5. Comparison of delay.

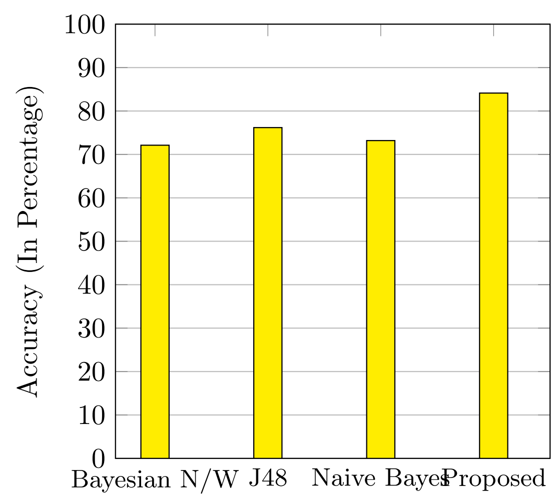

Fig. 6. Classifier accuracy.

5.2. Performance evaluation of proposed classifier

Performance evaluation of a classification model is an essential step since it shows how good is the classifier. This experiment aims to forecast cases using thyroid data, divided into training and test datasets. The dataset is divided into 66% for training and 34% for testing. Python language and its is artificial, and machine learning related libraries are utilized for data operation. For data preprocessing, rapid accessing to data, and intuitive data operation, Pandas is used. Scikit-learn is applied during model training and evaluation phases, which is also an open-source library providing comprehensive tools for Artificial Intelligence. Pandas and Scikit-learn are built on NumPy, SciPy, and Matplotlib, three entire libraries for scientific computing with Python. The experiments are conducting by utilizing unique datasets.

5.2.1. Confusion matrix

The confusion matrix establishes a link between the classifier's decisions and the labels assigned to the samples. It is a method for determining the classification system's consistency. The well-classified cases are located on the diagonal of the uncertainty matrix, while the remainder is misclassified. The matrix is an estimation parameter that is concerned with the accurate classification and distribution of various groups. Confusion matrices are used to provide additional information regarding the grouping of samples belonging to a specified class. We may quantify many statistical measures from a confusion matrix, including Sensitivity and Specificity.

TP: represent the number of hypothyroid-infected individuals classified as hypothyroid infected.

FP: represent the number of hypothyroid uninfected individuals classified as hypothyroid infected.

FN: represent the number of hypothyroid infected individuals classified as hypothyroid uninfected patients.

TN: represent the number of hypothyroid uninfected individuals classified hypothyroid uninfected.

The table shows the accuracy

5.2.2. Classification accuracy

Classification accuracy is the major criteria to evaluate a classifier. It shows the efficiency by calculating the percentage of true prediction. Fig. 6 clearly shows the proposed classifier gives the better results as compare to traditional classifiers (Fig. 7).

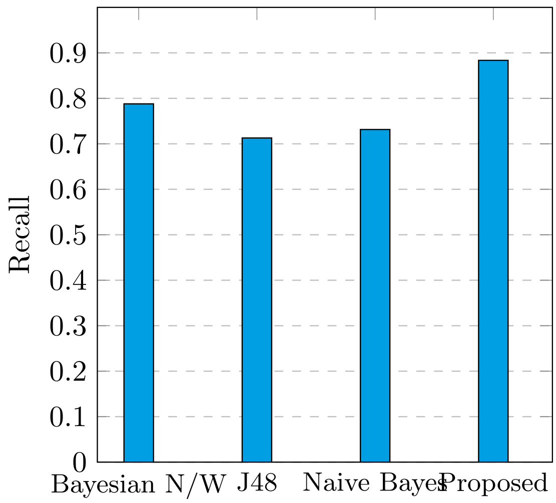

Fig. 7. Sensitivity.

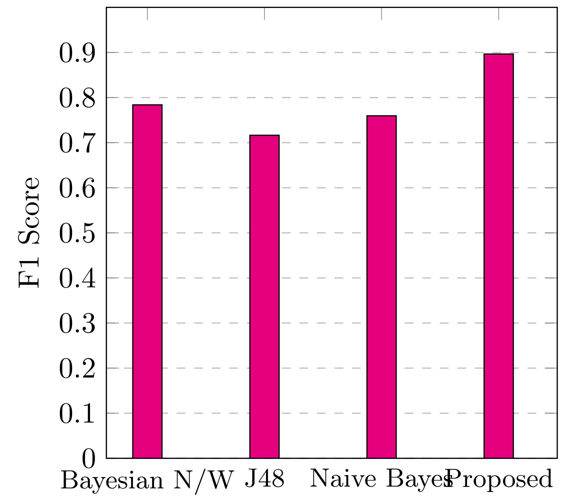

Fig. 9. F1 score.

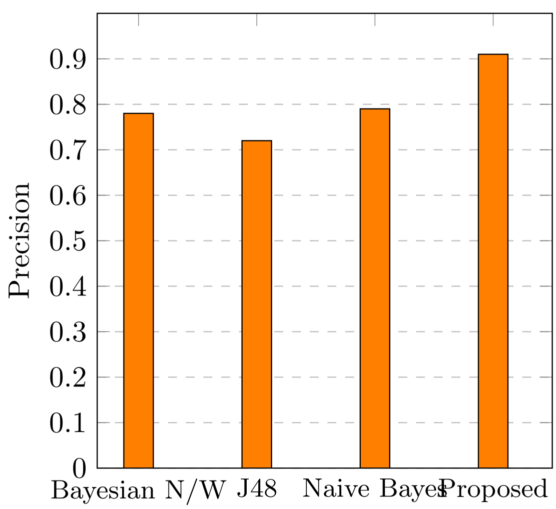

Fig. 8. Precision.

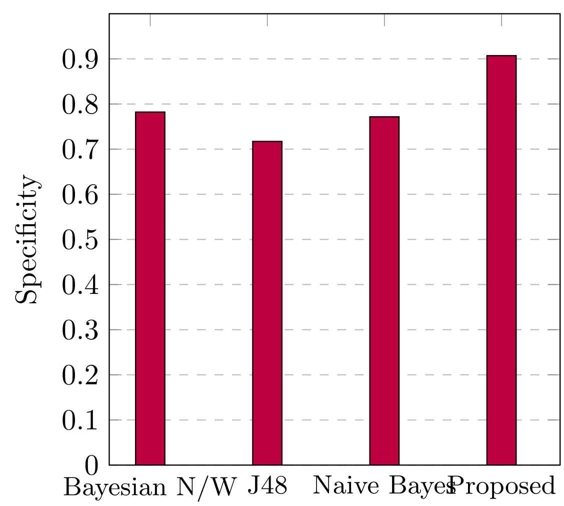

Fig. 10. Specificity.

5.2.3. Precision

Precision is the percentage of the true positive to the all positive values. Fig. 8 shows that proposed classifier is better as compare to other traditional classifier (Figs. 9 and 10).

5.2.4. ROC curve

ROC is a graphic representation system that measures the output of a binary classier on the one hand and measures the separate descriptors on the other hand. For several years, its use has become indispensable as the evaluation method of the decision support systems. Fig. 11 shows the ROC curve of the proposed classifier which shows how much the proposed classifier is capable of distinguishing between classes i.e. positive and negative.

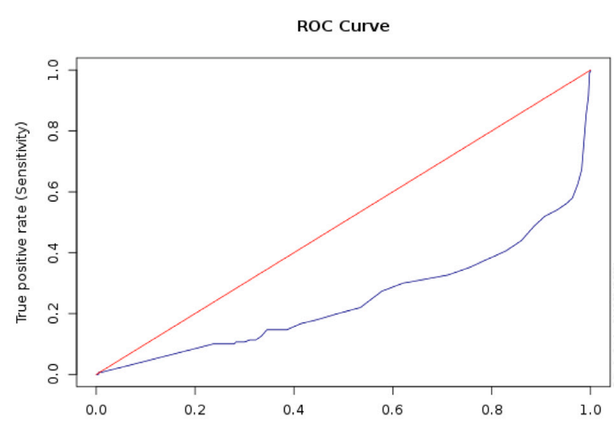

Fig. 11. ROC curve.

5.2.5. Sensitivity & specificity

Sensitivity quantifies the frequency at which a test accurately produces a favorable outcome in individuals that have the disorder being checked for . A highly responsive examination can detect nearly anyone that has the disease and can provide few false-negative outcomes (Fig. 12).

Specificity refers to a test's capacity to correctly produce a negative outcome in the absence of the condition being measured. A high-specificity examination accurately excludes nearly all who may not have the disorder and produces a small number of false-positive outcomes.

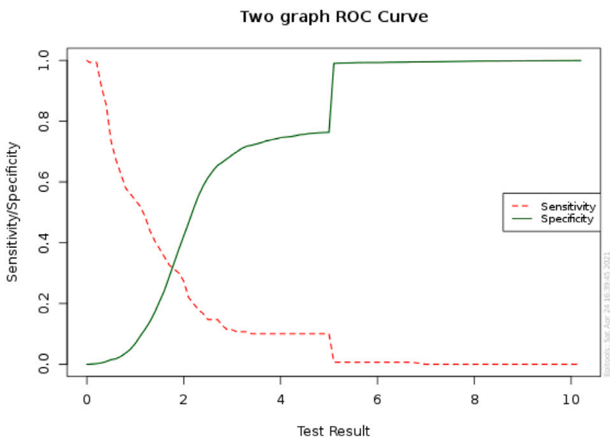

Fig. 12. Sensitivity & specificity.

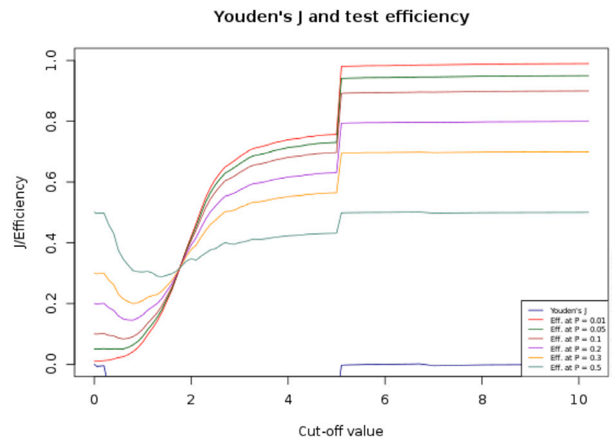

Fig. 14. Youden's J and test efficiency.

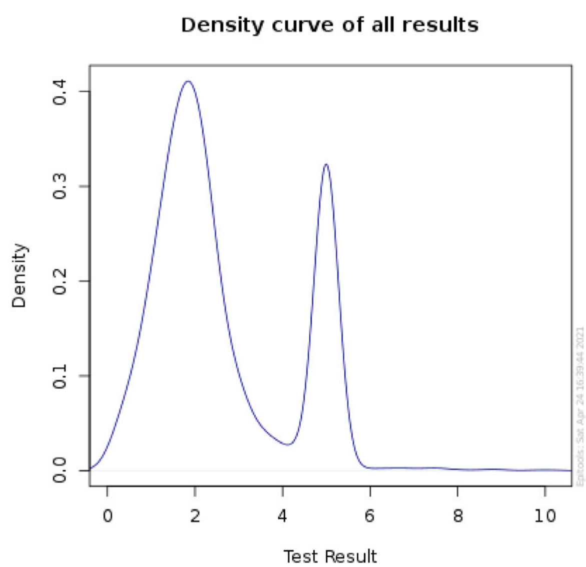

Fig. 13. Density plot.

5.2.6. Density plots

Density plots are another simple and fast strategy for obtaining the distribution of each attribute. Similar to a histogram, except that a smooth curve is drawn over the top of each bin. They are often referred to as abstracted histograms (Fig. 13).

5.2.7. Youden's J and test efficiency

Youden's J is the probability of a favorable test outcome in subjects with the disorder compared to those without. Youden's J index is a value between 0 and 1 that blends sensitivity and specificity into a single calculation (Sensitivity + Specificity - 1). Youden's index equals one in an ideal examination. It is also equal to the vertical distance between the ROC curve for a single judgment threshold and the diagonal no discrimination axis (Fig. 14).

6. Conclusion

The smart healthcare industry is still at an early stage; researchers are concerned both with processing capacity and latency. Thus, following a combined analysis of the key problems of smart health applications, a QoS aware of AI-assisted IoT-based safe healthcare as a service architecture for smart city applications is being proposed. Artificial intelligence is utilized to provide the system with intelligence. The proposed framework uses fog computing in QoS to reduce the latency, network cost, and energy consumption for each system-related device. In this paper, the technique for thyroid prediction at an early stage is improved by adding a classifier assembly strategy. The thyroid dataset includes 215 cases from the UCI machine study library (University of California - Irvine). On the other hand, this framework offers patients and their devices safe and permitted access via encryption and decryption. Two kinds of experiments are conducted. The findings demonstrate the improvement of the framework and demonstrate that it will effectively support intelligent health services and increase the quality of service.

Due to the ignored effect of adjustment of hyper-parameters in this research, the proposed framework may be further enhanced using optimization techniques. Soon a generic fog framework for many kinds of disease, including COVID-19, swine flu, AIDS, and cancer, may be created.

CRediT authorship contribution statement

Prabh Deep Singh: Conceptualization, Methodology, Software. **Gaurav Dhiman:** Visualization, Data curation. **Rohit Sharma:** Writing – original draft, Investigation, Writing – review & editing.

References

[1] P.E. Petersen, The World Oral Health Report 2003: continuous improvement of oral health in the 21st century–the approach of the WHO Global Oral Health Programme, Community Dent. Oral Epidemiol. 31 (2003) 3–24.
[2] R. Kumar, G. Dhiman, A comparative study of fuzzy optimization through fuzzy number, Int. J. Mod. Res. 1 (1) (2021) 1–14.
[3] I. Chatterjee, Artificial intelligence and patentability: Review and discussions, Int. J. Mod. Res. 1 (1) (2021) 15–21.
[4] P.K. Vaishnav, S. Sharma, P. Sharma, Analytical review analysis for screening COVID-19, Int. J. Mod. Res. 1 (1) (2021) 22–29.
[5] G. Dhiman, V. Kumar, Emperor penguin optimizer: A bio-inspired algorithm for engineering problems, Knowl.-Based Syst. 159 (2018) 20–50.
[6] G. Dhiman, V. Kumar, Multi-objective spotted hyena optimizer: a multi-objective optimization algorithm for engineering problems, Knowl.-Based Syst. 150 (2018) 175–197.
[7] G. Dhiman, V. Kumar, Seagull optimization algorithm: Theory and its applications for large-scale industrial engineering problems, Knowl.-Based Syst. 165 (2019) 169–196.
[8] G. Dhiman, V. Kumar, Knrvea: A hybrid evolutionary algorithm based on knee points and reference vector adaptation strategies for many-objective optimization, Appl. Intell. 49 (7) (2019) 2434–2460.
[9] B.D. Deebak, F. Al-Turjman, Smart mutual authentication protocol for cloud based medical healthcare systems using internet of medical things, IEEE J. Sel. Areas Commun. 39 (2) (2020) 346–360.
[10] A. Ullah, M. Azeem, H. Ashraf, A.A. Alaboudi, M. Humayun, N. Jhanjhi, Secure healthcare data aggregation and transmission in IoT—A survey, IEEE Access 9 (2021) 16849–16865.

[11] K.C. Okafor, I.E. Achumba, G.A. Chukwudebe, G.C. Ononiwu, Leveraging fog computing for scalable IoT datacenter using spine-leaf network topology, J. Electr. Comput. Eng. 2017 (2017).
[12] A. Asghar, A. Abbas, H.A. Khattak, S.U. Khan, Fog based architecture and load balancing methodology for health monitoring systems, IEEE Access 9 (2021) 96189–96200.
[13] U.U. Rehman, S.-B. Park, S. Lee, Secure health fog: A novel framework for personalized recommendations based on adaptive model tuning, IEEE Access 9 (2021) 108373–108391.
[14] A. Hussain, K. Zafar, A.R. Baig, Fog-centric IoT based framework for healthcare monitoring, management and early warning system, IEEE Access 9 (2021) 74168–74179.
[15] G. Muhammad, M. Alhussein, Convergence of artificial intelligence and internet of things in smart healthcare: A case study of voice pathology detection, IEEE Access 9 (2021) 89198–89209.
[16] G. Dhiman, A. Kaur, Stoa: a bio-inspired based optimization algorithm for industrial engineering problems, Eng. Appl. Artif. Intell. 82 (2019) 148–174.
[17] S. Kaur, L.K. Awasthi, A. Sangal, G. Dhiman, Tunicate swarm algorithm: A new bio-inspired based metaheuristic paradigm for global optimization, Eng. Appl. Artif. Intell. 90 (2020) 103541.
[18] M. Dehghani, Z. Montazeri, O.P. Malik, G. Dhiman, V. Kumar, Bosa: binary orientation search algorithm, Int. J. Innov. Technol. Explor. Eng. 9 (2019) 5306–5310.
[19] G. Dhiman, ESA: a hybrid bio-inspired metaheuristic optimization approach for engineering problems, Eng. Comput. (2019) 1–31.
[20] Z. Sun, H. Yin, H. Chen, T. Chen, L. Cui, F. Yang, Disease prediction via graph neural networks, IEEE J. Biomed. Health Inf. 25 (3) (2020) 818–826.
[21] S. Proto, E. Di Corso, D. Apiletti, L. Cagliero, T. Cerquitelli, G. Malnati, D. Mazzucchi, Redtag: a predictive maintenance framework for parcel delivery services, IEEE Access 8 (2020) 14953–14964.
[22] F. Shamout, T. Zhu, D.A. Clifton, Machine learning for clinical outcome prediction, IEEE Rev. Biomed. Eng. 14 (2020) 116–126.
[23] V. Kumar, D.R. Recupero, D. Riboni, R. Helaoui, Ensembling classical machine learning and deep learning approaches for morbidity identification from clinical notes, IEEE Access 9 (2020) 7107–7126.
[24] G. Dhiman, Moshepo: a hybrid multi-objective approach to solve economic load dispatch and micro grid problems, Appl. Intell. 50 (1) (2020) 119–137.
[25] G. Dhiman, M. Garg, Mosse: a novel hybrid multi-objective meta-heuristic algorithm for engineering design problems, Soft Comput. (2020) 1–20.
[26] G. Dhiman, Multi-objective metaheuristic approaches for data clustering in engineering application (s), (Ph.D. thesis), 2019.
[27] G. Dhiman, A. Kaur, HKn-RVEA: a novel many-objective evolutionary algorithm for car side impact bar crashworthiness problem, Int. J. Veh. Des. 80 (2–4) (2019) 257–284.
[28] G. Dhiman, K.K. Singh, A. Slowik, V. Chang, A.R. Yildiz, A. Kaur, M. Garg, EMoSOA: a new evolutionary multi-objective seagull optimization algorithm for global optimization, Int. J. Mach. Learn. Cybern. (2020) 1–26.
[29] G. Dhiman, D. Oliva, A. Kaur, K.K. Singh, S. Vimal, A. Sharma, K. Cengiz, BEPO: A novel binary emperor penguin optimizer for automatic feature selection, Knowl.-Based Syst. (2020) 106560.
[30] G. Dhiman, K.K. Singh, M. Soni, A. Nagar, M. Dehghani, A. Slowik, A. Kaur, A. Sharma, E.H. Houssein, K. Cengiz, MOSOA: A new multi-objective seagull optimization algorithm, Expert Syst. Appl. (2020) 114150.
[31] H. Kaur, A. Rai, S.S. Bhatia, G. Dhiman, MOEPO: A novel Multi-objective Emperor Penguin Optimizer for global optimization: Special application in ranking of cloud service providers, Eng. Appl. Artif. Intell. 96 (2020) 104008.
[32] T.M. Le, T.M. Vo, T.N. Pham, S.V.T. Dao, A novel wrapper–based feature selection for early diabetes prediction enhanced with a metaheuristic, IEEE Access 9 (2020) 7869–7884.
[33] P. Sun, D. Wang, V.C. Mok, L. Shi, Comparison of feature selection methods and machine learning classifiers for radiomics analysis in glioma grading, IEEE Access 7 (2019) 102010–102020.
[34] R. Ghorbani, R. Ghousi, A. Makui, A. Atashi, A new hybrid predictive model to predict the early mortality risk in intensive care units on a highly imbalanced dataset, IEEE Access 8 (2020) 141066–141079.
[35] H.L. Zhang, Y. Zhao, C. Pang, J. He, Splitting large medical data sets based on normal distribution in cloud environment, IEEE Trans. Cloud Comput. 8 (2) (2015) 518–531.
[36] Y. Zhang, X. Wang, N. Han, R. Zhao, Ensemble learning based postpartum hemorrhage diagnosis for 5G remote healthcare, IEEE Access 9 (2021) 18538–18548.
[37] H. Gupta, A. Vahid Dastjerdi, S.K. Ghosh, R. Buyya, iFogSim: A toolkit for modeling and simulation of resource management techniques in the Internet of Things, Edge and Fog computing environments, Softw. - Pract. Exp. 47 (9) (2017) 1275–1296.