IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 4 August 2023, accepted 28 August 2023, date of publication 6 September 2023, date of current version 20 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3312343

RESEARCH ARTICLE

A Clinical Decision Support System for Edge/Cloud ICU Readmission Model Based on Particle Swarm Optimization, Ensemble Machine Learning, and Explainable Artificial Intelligence

MAALI ALABDULHAFITH[1], HAGER SALEH[2], HELA ELMANNAI[1], ZAINAB HASSAN ALI[3], SHAKER EL-SAPPAGH[4,5], JONG-WAN HU[6,7], AND NORA EL-RASHIDY[8]

[1]Department of Information Technology, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia
[2]Faculty of Computers and Artificial Intelligence, South Valley University, Hurghada 83523, Egypt
[3]Department of Embedded Network Systems Technology, Faculty of Artificial Intelligence, Kafrelsheikh University, Kafr El-Sheikh 13518, Egypt
[4]Faculty of Computer Science and Engineering, Galala University, Suez 435611, Egypt
[5]Information Systems Department, Faculty of Computers and Artificial Intelligence, Benha University, Banha 13518, Egypt
[6]Department of Civil and Environmental Engineering, Incheon National University, Incheon 22012, South Korea
[7]Incheon Disaster Prevention Research Center, Incheon National University, Incheon 22012, South Korea
[8]Machine Learning and Information Retrieval Department, Faculty of Artificial Intelligence, Kafrelsheikh University, Kafr El-Sheikh 13518, Egypt

Corresponding authors: Hager Saleh (Hager.Saleh@fcih.svu.edu.eg) and Jong-Wan Hu (jongp24@inu.ac.kr)

This research was supported by the Basic Science Research Program through a National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (NRF-2021R1A2B5B02002599) and Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2023R407), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

ABSTRACT ICU readmission is usually associated with an increased number of hospital death. Predicting readmission helps to reduce such risks by avoiding early discharge, providing appropriate intervention, and planning for patient placement after ICU discharge. Unfortunately, ICU scores such as the simplified acute physiology score (SAPS) and Acute Physiology and Chronic Health (APACHE) could help predict mortality or evaluate illness severity. Still, it is ineffective in predicting ICU readmission. This study introduces a clinical monitoring fog-computing-based system for remote prognosis and monitoring of intensive care patients. This proposed monitoring system uses the advantages of machine learning (ML) approaches for generating a real-time alert signal to doctors for supplying e-healthcare, accelerating decision-making, and monitoring and controlling health systems. The proposed system includes three main layers. First, the data acquisition layer, in which we collect the vital signs and lab tests of the patient's health conditions in real-time. Then, the fog computing layer processes. The results are then sent to the cloud layer, which offers sizable storage space for patient healthcare. Demographic data, lab tests, and vital signs are aggregated from the MIMIC III dataset for 10,465 patients. Feature selection methods: Genetic algorithm (GA) and practical swarm optimization (PSO) are used to choose the optimal feature subset from detests. Moreover, Different traditional ML models, ensemble learning models, and the proposed stacking models are applied to full features and selected features to predict readmission after 30 days of ICU discharge. The proposed stacking models recorded the highest performance compared to other models. The proposed stacking ensemble model with selected features by POS achieved promising results (accuracy = 98.42, precision = 98.42, recall = 98.42, and F1-Score = 98.42), compared to full features and selected features. We also, provide model explanations to ensure efficiency, effectiveness, and trust in the developed model through local and global explanations.

INDEX TERMS Machine learning, ensemble learning, stacking ensemble learning, ICU readmission rate, explainable artificial intelligence.

I. INTRODUCTION

An intensive care unit (ICU) is a hospital department that constantly supervises patients with life-threatening health

The associate editor coordinating the review of this manuscript and approving it for publication was Vivek Kumar Sehgal.

conditions or injuries. Unplanned hospital readmissions are a dangerous indication for patients and an unnecessary waste of medical resources. To address the issue of unplanned readmissions, the Affordable Care Act (ACA) established the Hospital Readmissions Reduction Program in 2010 to impose penalties on hospitals with higher-than-expected 30-day readmission rates [1]. According to data released by the Centers for Medicare & Medicaid Services (CMS), hospitals have received nearly $2.5 billion in penalties for readmissions since the program's inception in 2012. This includes an estimated $564 million in 2018, $144 million more than in 2016. Readmission is also associated with adverse patient health outcomes, including increased mortality and length of stay. Accordingly, it is essential to identify which patients are ready for discharge. Patients at high risk of clinical deterioration may be beneficial and improve the outcomes of critically sick hospitalized patients.

The decision to discharge the patient from the ICU should consider several issues. Recently, hospitals have become more concerned with patient electronic health records (EHRs) data. Patient EHR includes various health information (demographics, vital signs, lab tests, clinical notes, and medications). The development of computational techniques contributes to enhancing clinical decision prediction. In this regard, machine learning (ML) has been applied to several issues in the ICU, such as mortality prediction [2], [3], sepsis prediction [4], cardiac arrest prediction [5], acute kidney injury [6], etc. In terms of predicting ICU readmission, several studies predict ICU readmissions based on available data upon discharge. For example, In [7], the authors predicted patients at risk and the risk of readmission based on patients' data until release. The study achieved promising results. However, earlier predictions are required for better accuracy and the abovementioned challenges. Other studies developed models to predict ICU readmission within 4872 hours of discharge and at any point during the hospital stay [8], [9], [10], [11], [12]. According to these studies [13], [14], the usual ICU readmission rate within 72 hours ranges from 4% to 11%.

The potential for utilizing complete EHR data has been investigated in several studies. For example, in [15], the authors used demographic data and lab tests of patients with heart failure to predict readmission based on ML techniques, concluding adequate performance in different evaluation metrics. Others utilized vital signs and ICU scores to predict readmission after 30 days of discharge [16].

Despite the promising results that have been achieved in the literature, most of them only consider the prediction model's performance. None of them consider the challenges related to the ICU shortage and how we could continuously monitor patients with a high risk of deterioration outside the ICU. To address this gap, we propose a complete monitoring system concerned with the speed of decision-making in intensive care, which plays a critical role in saving many human lives and is mainly based on the high quality of data processing. Thus, Fog technology as a modern computing platform has provided speedy and timely decision-making [17]. While using fog computing technology in the healthcare domain, primarily with time-sensitive real-time applications, fog computing ensures that no redundant data is sent to the cloud server to conserve network bandwidth usage and reduce transmission delay and data processing time [18].

One of the primary benefits of using fog computing in medical systems is the ability to speed up the operation of decision-making. Applying machine learning (ML) models as close to end-users can greatly reduce the time required to process and analyze data. This can be especially critical in emergencies where time is of the essence. Fog computing can also increase the ability to track data from everywhere with low delay time. This is due to its ability to support the feature of location awareness, unlike cloud computing. By enabling the collection and analysis of real-time data from various sources, healthcare professionals can better understand patient needs and provide more personalized care [19].

Another significant benefit of fog computing in medical systems is the reduced network bandwidth usage. By processing data locally, without the need to travel this data to the cloud server, fog computing reduces network bandwidth usage and improves system performance. This technique also increases the data delivery ratio, making the system more accurate and reliable. Finally, integrating fog technology, IoT, and clinical systems is expected to improve medical services and reduce hospital costs. By enabling remote patient monitoring and reducing the need for patients to be in the hospital continuously, fog computing can improve patient outcomes while lowering costs [20], [21].

Despite the promising results that have been achieved in the literature, most of them only consider the prediction model's performance. None of them consider the challenges related to the ICU shortage and how we could continuously monitor patients with a high risk of deterioration. To address this gap, we propose three main research questions to identify the most features to predict ICU readmission over time

1) Which features have a significant effect in predicting readmission ?
2) Can the utilization of multiple machine learning models in developing an ensemble model lead to a more robust model with superior performance compared to using a single machine learning model ?
3) Could we utilize cloud/ fog technologies to monitor patients outside ICU continuously ?

Our proposed system's main contribution can be summarized in the following points.

- Our proposed approach uses an ensemble model to predict a patient's readmission to ICU after discharge. The proposed system leverage fog/cloud technology to create continuous monitoring systems.
- To our knowledge, no existing studies in the literature have examined readmission prediction by considering continuous monitoring using fog/cloud technologies and specific features to make an accurate prediction.

- Utilizing meta-heuristic optimization techniques, such as Genetic Algorithm (GA) and Particle Swarm Optimization (PSO), to identify the optimal feature subset for our system.
- Investigating the impact of single and diverse machine learning (ML) and ensemble ML models in improving readmission prediction.
- Extending the proposed framework by applying explanation tools to provide an understandable explanation of the developed model

Our study was conducted in collaboration with medical experts, who helped us select the initial set of relevant features that are medically significant. We implemented and tested our model using a large population from the MIMIC-III dataset.

The rest of the paper could be summarized as follows: related work is detailed in Section II. The proposed clinical monitoring fog-Computing-based System is described in section III. The machine learning model based on the fog node is described in section IV. Section V shows the results. Discussion and comparison with other work are shown in section VI, and the paper concludes in section VIII.

II. RELATED WORK

The issue of hospital readmissions has been addressed in a number of studies [22], [23], [24]. Among the numerous works developed to provide supportive decisions in ICU is the prediction of mortality, LOS, and ICU readmission. This section summarizes the state of the art that utilized ML and DL to predict ICU readmission inside and outside hospitals.

First, some studies utilized ML to predict the readmission probability after the patients move from ICU to award in the same hospital (same visit).

In [25], the authors explored the performance of eight classification algorithms, including NB, DT, rule-based approaches, and ensemble methods applied to various sets of attribute data, and the effectiveness of each algorithm was assessed using six criteria. There were 185 variables in the 9926 individuals analyzed in the dataset from three ICUs in a Brazilian university hospital. These included demographics, duration of stay before admission to the ICU, comorbidities, severity indexes, treatments, organ support care during ICU stay, and laboratory results. In [22], the authors applied A boosted C5.0 tree as the base classifier and combined it with SVM as a secondary classifier to predict ICU. 20321 anonymized admission data were used to induce and validate the models at Veterans Health Administration (VHA) hospitals in Pittsburgh between fiscal years 2006 and 2014. In [26], ML models were predicted to be differentiated between patients who were unplanned ICU readmittances or deaths from those who were not. The models trained on 3326 ICU episodes that were collected from MMIC and tested on the target hospital's data. In [14], the authors used a gradient-boosted model based on real-time risk scores to predict readmission using 24885 patients collected from hospitals in the United States (US). The gradient-boosting ML mode performed the best since each DT is sequentially trained. In [27], Rule-based algorithms: DT, RF, XGBoost, and LASSO models were used using the 2016 Nationwide Readmissions Database (NRD) database from a US hospital. Data resampling techniques were used to provide more balanced data to effectively address the readmission and non-readmission groups. In [28], the authors selected the most important features from the dataset using the LASSO feature selection algorithm. Then the authors studied the effect of applying ML models to predict readmission based on the selected features. A hospital registry system was used to collect data on 5791 COVID-19 hospitalizations retrospectively. In [29], LR, DT, SVM, XGB, extremely randomized trees (ET), light gradient boosting machine (LGBM), and extreme gradient boosting (XGB) were used to predict ICU readmission using 58036 patients were collected from hospitals in Southeast Asia. LR recorded the highest AUC. In [30], the authors applied ML models with different feature selection methods to predict ICU mortality risk. A large hospital in Anhui provided data for 1628 patients with cardio-macrovascular disease in the ICU. In terms of performance, RF achieved the best results. In [31], the authors applied fuzzy modeling with tree search feature selection using 26655 patients from MIMICII to predict ICU readmission between 24h and 72h. The results showed that fuzzy modeling achieved the highest performance. In [32], the authors applied LR, SVM, RF, GBDT, and NB using text data for 40000 patients from MIMICII. They represented features using Bag-of-words and Term frequency-inverse document frequency (tf-idf). The results showed that LR achieved the highest AUC.

Others developed a deep-learning model to predict ICU readmission. In [33], the authors proposed heterogeneous long short-term memory (LSTM) and convolutional neural network (CNN) (CNN-LSTM) architecture for readmission risk measurement using the MMIC dataset. The architecture combined three stacked bidirectional LSTM layers, a CNN layer, and a dense decision layer. These layers were preceded by a preprocessing stage of categorical encoding using a MIMIC-III database. In [34] to incorporate the multivariate aspects of EHRs and catch abrupt and sudden fluctuations in chart event features like glucose and heart rate. The readmission prediction model was developed using three categories of features: chart events, ICD-9 embeddings, and patient demographic data. The authors conducted deep-learning models using the MIMIC-III dataset. They studied each feature that contributed to the prediction model over ten years. In [35], the authors applied CNN models and RF to the discharge summary notes in the MIMIC database and trained models were utilized to categorize and identify potentially high-risk admissions/patients. The CNN approach achieved the highest f1-score.

III. PROPOSED CLINICAL MONITORING FOG-COMPUTING-BASED SYSTEM

The main objective of this section is to introduce a clinical monitoring fog-computing-based system for remote prognosis and monitoring of the state of intensive care patients using their health symptoms. This proposed monitoring system uses the advantages of ML approaches for generating a real-time alert signal to doctors for supplying e-healthcare, accelerating decision-making, and monitoring and controlling the health systems remotely. As shown in Figure 1, the fog-computing-based system consists of three proposed layers: data acquisition, fog computing, and cloud server. To avoid delay time and accelerate decision-making, the proposed ML model has deployed on the fog nodes residing at the fog computing layer instead of the traditional data transferring methods that use straightforward data uploading to the cloud server. The fog nodes hold an enhanced performance using the practical swarm optimization algorithm. This optimizer can select the most vital signs according to the nature of each pathological condition within the care unit. So, it helps medical experts to speed up decision-making with higher accuracy and faster convergence.

A. DATA ACQUISITION LAYER

In this layer, a set of IoT devices called medical sensors are scattered to collect the vital signs of the patient's health conditions in real-time. These IoT devices are categorized as class-I for the lowest potential risk and class-IV for the highest potential risk. To sense the health status of the extensive care patients, the medical sensors obtain the vital signs such as hemoglobin, heart rate, heart rhythm, sodium bicarbonate, GCS total, glucose, blood pressure, WBC, urea nitrogen, protein percentage, and oxygen through a set of embedded sensors such as the temperature sensor, EEG, and ECG are along with wearable devices in which these medical sensors have been designed to be comfortably utilized by the patients. All these collected data will be traveled to the Fog computing layer via wireless network technologies such as Bluetooth, WIFI, and Radio-Frequency Identification (RFID). The communication range for these technologies is described in Table 1.

TABLE 1. The communication range of communication technologies.

Technology	Bit rate	Distance	Security	Comm. Range
Bluetooth	2.1 Mbps	20-200 m	128 bits AES	2.4 GHz
WIFI	1 M to 300 Mbps	50 m	High	13.56 MHz
RFID	106 K to 424 Kbps	20 cm	High	2.4 G-5 GHz

B. THE FOG COMPUTING LAYER

It supports the numerous scattered nodes known as gateways. The gateways are distributed computing, networking, and storage equipment near the medical sensors. These sensors are in charge of gathering the data and recording the happenings. This layer makes four processes easier: receiving sensor data, analyzing it for health-related decisions, alerting the carers, and storing it all on a cloud server. A local data processing system has been added to fog computing to strengthen its intelligence. This has increased decision-making speed, decreased latency, prevented network connectivity problems, and improved system dependability. The following list of the fog computing layer's characteristics is provided:

- The data acquisition process mainly relies on embedded IoT devices such as the temperature sensor, EEG, and ECG, along with wearable devices, which these medical sensors have been designed to utilize by the patients comfortably. After these vital signs have been collected, the preprocessing stage will be performed.
- In the process of data analysis, the data gathered from the blood pressure, brain, and temperature sensors are kept in the fog computing layer, where it is analyzed for any emergency conditions depending on age, height, and disease using the optimization method (Particle Swarm) and machine learning techniques. The workload on a fog computing device (fog computing node) and its location and behavior change often in fog computing environments. An unstable and extremely dynamic fog computing architecture makes it possible for a fog node to be trustworthy at one point and unreliable at another. The medical sensor devices typically send many service requests to the fog computing layer, which the fog nodes should handle. However, a fog node's resources could not be sufficient to provide the requested service, resulting in a QoS violation, a delay in response, or even service failure sometimes. Therefore, improving the fog node performance using the integration between the practical swarm optimization and machine learning technique is significant, which is described in Section IV.
- In system notification, clinicians become distracted and ignore potentially fatal reports due to the alarm overload and continual acoustics brought on by loud in-room and remote workstation medical device alarms. Alert weariness is a syndrome that occurs when doctors become desensitized to excessive alert stimuli, potentially delaying medical intervention and care. The goal of clinical alarm management has typically been to minimize nuisance alerts or alarms that cannot be clinically addressed.

C. CLOUD COMPUTING LAYER

The servers, repositories, and distributed resources make up this layer. The cloud server oversees every device linked to the cloud layer, making receiving, processing, and storing patient data easier. This information can be utilized to examine the patient's medical status in ICU. The cloud layer's characteristics are generally listed as follows:

- Data storage: following the fog computing layer's data analysis process, the results are sent to the cloud layer, which offers a sizable storage space for keeping patient healthcare data for further study by carers, physicians, hospitals, and insurance providers.

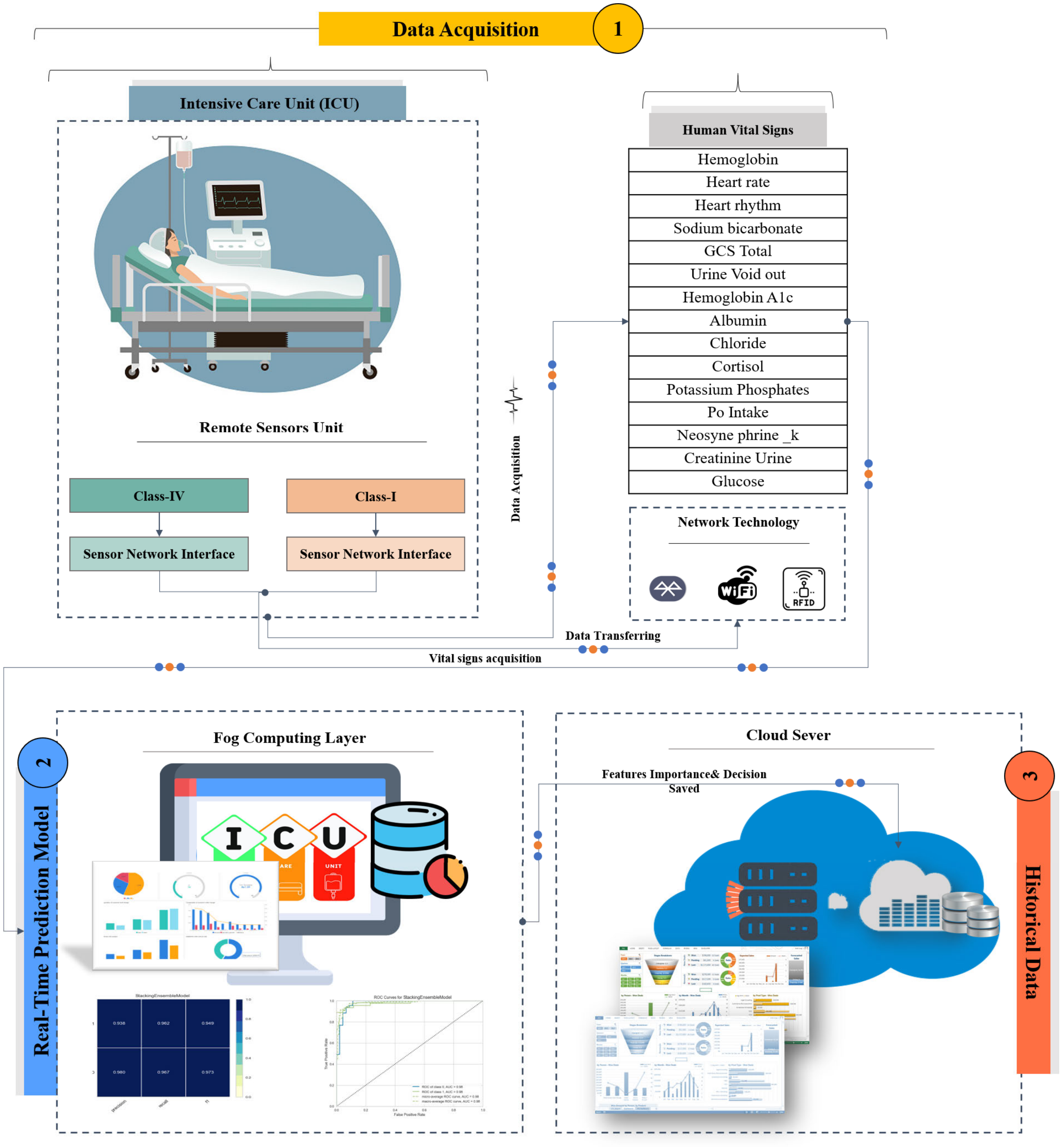

FIGURE 1. Proposed clinical monitoring fog-computing-based system.

- Data analysis: for upcoming studies in clinical decision-making, the patient's health data, which is kept in the cloud and includes pictures of the diseased portions, descriptions of the symptoms, therapies, and therapy plans, is analyzed deeply. Many machine learning algorithms and data visualization methods could be used to understand these data better.

D. THE DEPLOYMENT OF FOG TECHNOLOGY INSIDE THE CLINICAL SYSTEM (SCALABILITY, PERFORMANCE)

Fog computing is a technology that has gained significant attention in recent years due to its ability to enhance cloud computing performance and support IoT technology. In clinical systems, fog computing has several advantages, including scalability and performance. One of the primary benefits of fog computing is its scalability. Unlike traditional systems, fog computing enables the distribution of computing resources across the network, including edge devices, gateways, and cloud servers. This allows for more efficient resource utilization and can improve system performance.

Fog computing can support the deployment of edge devices, providing additional computing resources and enhancing the system's scalability. These devices can perform tasks such as data processing, filtering, and analysis, reducing the burden on the cloud servers and improving the system's overall performance. Fog computing can also enhance the performance of clinical systems. By processing data locally, fog computing reduces the need to transfer large amounts of data to the cloud server, which can result in significant latency reduction. This can be especially critical when real-time data analysis is required, such as in emergencies.

Furthermore, fog computing can support the deployment of ML models close to end-users, improving the accuracy and speed of decision-making. This can be especially important in clinical systems where accurate and timely decision-making is critical to patient outcomes. Fog computing also has several benefits for clinical systems, including scalability and performance. By distributing computing resources across the network and supporting the deployment of edge devices, fog computing can improve the system's scalability. Furthermore, by processing data locally and supporting the deployment of ML models close to end-users, fog computing can significantly enhance the performance of clinical systems. Overall, fog computing is a promising technology that can revolutionize the healthcare industry and provide better care for patients.

IV. MACHINE LEARNING MODEL BASED ON FOG NODE

The primary steps that determine the proposed ML model procedure are discussed in this section. The proposed ML model is located on the fog nodes residing at the fog computing layer. They perform enhanced using optimization selection methods such as the PSO and GA algorithms. These methods can select the most vital signs according to the nature of each pathological condition within the care unit. Therefore, it helps medical experts to make decisions with higher accuracy and faster convergence. Generally, using the fog computing layer to deploy our proposed ML model significantly reduces delay processing and high transmission time because the fog computing layer is installed close to the end user, unlike the cloud server.

Figure 2 shows the main steps of developing ML models to predict ICU. The main steps include data description, preprocessing data, optimization selection methods, ML models, ensemble models, and stacking models.

A. DATA DESCRIPTION

Medical Information Mart Intensive Care III is a benchmark dataset created by MIT [36]. It includes EHR data for 53,432 patients admitted to different ICUs (SICU, MICU, CRCU, CSRU, etc.). A total of 4750 measurements and 470 laboratory tests are included in MIMIC III tables. In our study, we only consider adult patients (age>15) which admitted to both Surgical ICU(SICU) and surgical recovery (SR ICU). All data is stored as comma-separated values (CSV) files that could be converted to a relational database through PostgreSQL. Table 2 details the tables used in the current study. A total number 10,465 patients were included in this study after applying the inclusion and exclusion criteria that could be summarized as follows; (1) exclude patients aged less than or equal to 15 years, (2) exclude patients that died inside ICU, (3) exclude patients who are admitted inside ICU rather than SICU and SRICU (4) exclude patients that did not have at least one value in each measurement (5) exclude some patients from making the data balanced.inclusion and exclusion criteria detailed in Figure 3.

B. PRE-PROCESSING

Several challenges appeared at this stage and heavily affected model performance, including missing values, removing outliers, and data imputation.

- Irregular time interval: An irregular interval sampled time series is a series of samples with irregular intervals between observations. When the time gaps between observations are wide, this data introduces a time sparsity factor. Most machine learning techniques take observation order into account since they need more time comprehension. As a result, it is more challenging to learn the time dependencies included in time series problems [37], [38]. With MIMIC-III, vital signs are measured irregularly. Some took measurements every few minutes, whereas others took measurements every few seconds. Some ML methods could handle time series data when sampled with the same interval unless most are not equipped to handle it. To address this issue, we averaged all measures throughout an hour to produce a single record of the patient's vital signs observations. As a result, each record has consistent values.
- Outlier removal is defined as values that deviate too much from the norm and are considered outliers [39], [40]. According to the judgments of medical experts, a normal range is defined. To identify outliers, first, we specify the normal range for each feature according to the medical expert, then remove the data that considers outliers from the medical expert's perspective.
- Filling missing value: a dataset's missing values are typically replaced by new values reasonably similar to those that would have been recorded if they had been observed. This procedure is known as data imputation.

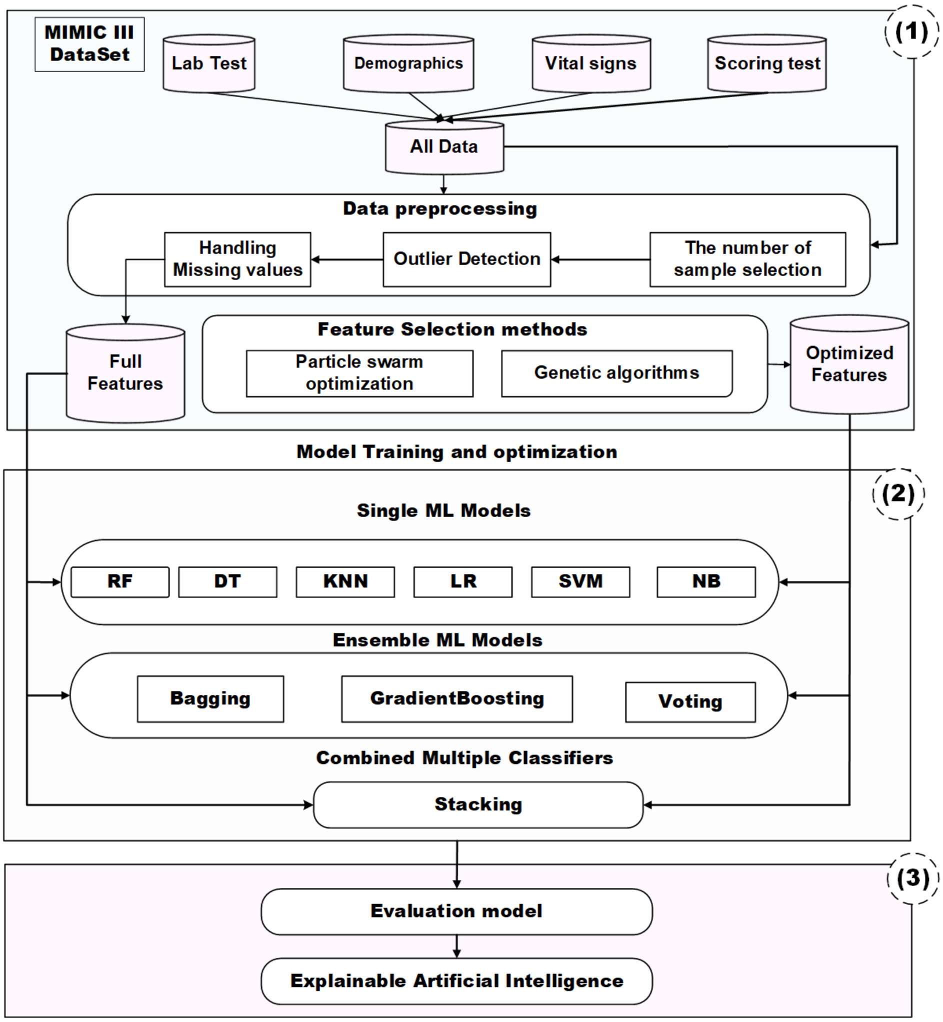

FIGURE 2. The architecture of the proposed system.

TABLE 2. MIMIC III Dataset description.

Table Name	Table description	Utilized columns
Patients	Include all patients' demographics (i.e., Subject, weight, BMI, Date of birth, etc.)	Subject, Weight, BMI, Date of birth, Expire Flag)
Char-Events	Include all vital signs recorded during ICU stay (i.e., Heartrate, blood pressure, etc.)	Item_ID, Chart time, and value
Lab-Events	Include all lab tests recorded during ICU stay (i.e., BUN, Albumin level, etc.)	Chart_time, Item_ID, Value
ICU-stay	Include all details related to the admitted care unit, such as (i.e. admission time, discharge time, etc.)	ICU_Stay ID, LOS, Addmission_time, discharge_time
D-lab-times	A dictionary that includes all details about measurement and lab tests	Label, Unit of measurement

Imputation techniques can often be divided into two groups: single imputation and multiple imputation. Single imputation substitutes plausible values for missing values only once, whereas multiple imputation methods perform imputation many times to produce a range of imputed values for each missing item. This gives information about the degree of uncertainty in the imputed values for imputation techniques with some stochastic elements. Missing values are frequently present in medical data. This raises several factors, such as sensor failure, data recording at various times, etc. Removing missing values is the most straightforward technique to handle them. However, doing it this way could result in losing important information. As a result, numerous

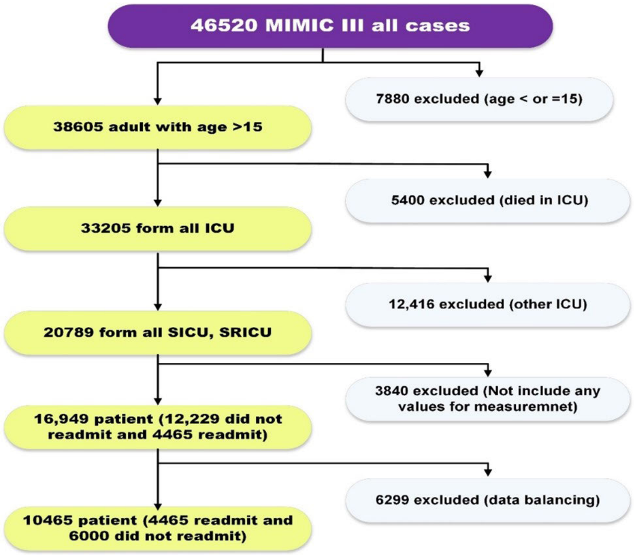
46520 MIMIC III all cases
7880 excluded (age < or =15)
38605 adult with age >15
5400 excluded (died in ICU)
33205 form all ICU
12,416 excluded (other ICU)
20789 form all SICU, SRICU
3840 excluded (Not include any values for measuremnet)
16,949 patient (12,229 did not readmit and 4465 readmit)
6299 excluded (data balancing)
10465 patient (4465 readmit and 6000 did not readmit)

FIGURE 3. Inclusion and exclusion criteria.

methods, such as expectation maximization [41] and hot deck encoding [42], have been devised to infer missing values based on the other records. Large portions of data (40-55%) in crucial features were lost during the preprocessing step of MIMIC-III, but we could not eliminate them because they were essential to the prediction process. In light of this, we chose scenarios where each measurement has at least two values before using expectation maximization [43] to impute further missing values.

C. FEATURE SELECTION OPTIMIZATION

Utilizing feature selection is a critical process that affects the model's overall performance. In this study, we utilized a genetic algorithm and PSO for feature selection (meta-heuristic feature selection). Choosing these two algorithms is owed to several reasons that could be summarized as follows. (1) PSO and Genetics could handle continuous and categorical variables [44]. Therefore, it doesn't require any assumptions about the underlying distribution of the data and can handle both discrete and continuous variables [45]. (2) Both PSO and GA can be easily parallelized, which means they can be run on multiple processors or computers to speed up the search process [46] (3) both consider population-based algorithms, which could handle high-dimensional datasets with a large number of features, the following subsections details the utilized techniques

D. PRACTICAL SWARM OPTIMIZATION (PSO)

The practical swarm optimization scenario is designed to select the optimal features (most vital signs) that directly affect the decision of the machine learning model. Figure 4 shows the flowchart of the standard PSO. Practical swarm optimization (PSO) is a meta-heuristic feature optimization algorithm inspired by swarms. It was developed to find the optimal solutions among solution spaces [47]. The mathematical model of PSO could be summarized in the following points (1) each practice has a position, fitness value, and velocity. (2) each practical search for the optimal fitness value and position. (3) a list of the best position and best fitness is recorded [48].

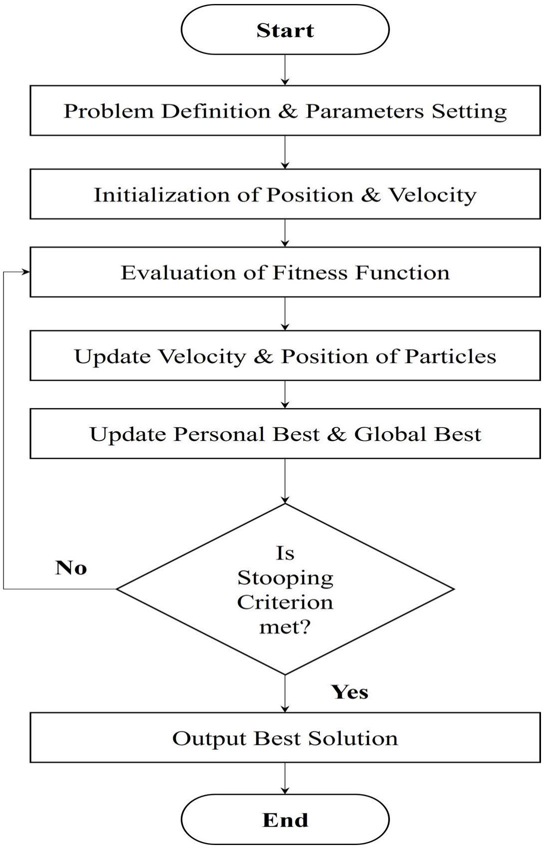
Start
Problem Definition & Parameters Setting
Initialization of Position & Velocity
Evaluation of Fitness Function
Update Velocity & Position of Particles
Update Personal Best & Global Best
Is Stooping Criterion met?
No
Yes
Output Best Solution
End

FIGURE 4. Flowchart of practical swarm optimization Algorithm.

E. GENETIC ALGORITHMS (GA)

GA is one of the enormously advanced evolutionary algorithms for extracting and selecting appropriate features. This randomized method for optimizing functions is based on biological evolution and natural genetics concepts [49], [50]. GA has been proven capable of exploring a huge space of solutions and dealing with many input characteristics. GA encodes the optimization function as bit arrays that mimic chromosomes, and Genetic operators customize strings to find a close-to-optimum answer to the situation at hand. This is performed using the method depicted in the following steps as in Figure 5 [51]:

- Coding the goals or cost functions.
- Establishing a fitness function.
- A generation is creating a population of individuals (solutions).

- Evaluating individuals in the population's fitness function.
- Forming an entirely novel population through crossover and mutation, fitness-proportionate growth, and then substituting the old population and looping with the new population.
- Decoding the outcomes to resolve the problem.

The main steps of GA are shown in Figure 5. The three primary genetic operators in genetic algorithms are crossover, mutation, and selection [51]:

- Crossover: It is the swapping parts of a solution represented by chromosomes or other solutions. The main purpose is to provide convergence and solution mixing in a subspace.
- Mutation: It is the haphazard modification of a single solution's constituent parts, which increases population diversity and offers a way to elude a local optimum.
- Selection of the fittest: it is the practice of transmitting alternatives with high fitness to future generations, which is usually done in the form of some selection of the most suitable solutions.

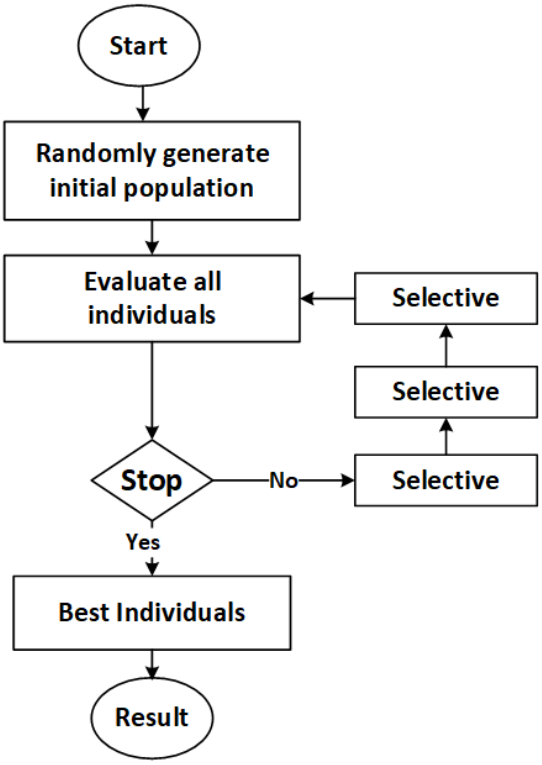

FIGURE 5. Genetic algorithm streaming steps.

F. BASELINE ML MODELS

Different baseline ML models: Random forest (RF) [52], [53], Logistic regression (LR) [54], [55], Decision tree (DT) [56], [57], Support vector machine (SVM) [58], [59], K-nearest neighbor (KNN) [60], and Naive Bayes (NB) [61], [62] are applied to full features and selected features to predict UCI. ML models are optimized using grid search and compared with the proposed stacking model.

G. ENSEMBLE LEARNING

Ensemble learning is a learning approach that aims to achieve enhanced performance by combining predictions from several models. These models are known as base models. Each model solves single-model challenges such as low accuracy, high variance, and feature bias. The weak learner could be from the same type of various types, and each weak learner maps the features input to the final output with different decision boundaries [63]. The three main types of ensemble learning are bagging, boosting, hard voting, and stacking. There are differences in the learning and the aggregating results; these methods can be summarized in the following points [64]. Bagging: the main idea in bagging is to build several models, then average the whole prediction; bagging is used to reduce variance and overfitting. Boosting: the main idea is to add several models sequentially, while each model concentrates on the errors produced by the previous model, out a weighted average of the total prediction. Stacking involves utilizing several models in the base learning and another model to combine the prediction [65], [66].

H. THE PROPOSED STACKING MODEL

Stacking involves utilizing several models in the base learning and another model to combine the prediction [65].

Write a description of the proposed ensemble model here

Algorithm 1 Stacking Model

```
Input: MIMIC III Dataset (Demographics, lab tests,
       vital signs)
Output: Classification readmission (Yes, No)
Extract statistical features (Min, Max, average) from
  numeric features (first, last) from categorical.
Apply particle swarm optimization to extract the
  optimal feature subset.
Divide Ds for training and testing data (70% for
  training and 30% for testing)
Define P models for the required task $N_k$ where k=
  1,2,3, p (level 0 learning)
for $k \leftarrow 1$ to $N$ do
    Pre-train model $N_k$ on training data (RF, DT,
      KNN, SVM, LR, and NB) // level 1 learning
    Save the model $N_k$
for $k \leftarrow 1$ to the number of saved models do
    Load the model
    Add meta classifier (RF) // level 2 learning.
    Use test data to evaluate the data using the saved
      model.
    The output of the level-0 learning based to level-1
      learning (meta-learning)
Return Give the final prediction.
```

I. EVALUATING THE MODELS

The accuracy, precision, recall, and F1-score of the models were used to evaluate the models. TP stands for true positive, TN stands for true negative, FP stands for false positive, and FN stands for false negative. Equations 1-4 can be found here.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}. \quad (1)$$

$$Precision = \frac{\mathrm{TP}}{\mathrm{TP + FP}} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

$$F1 = \frac{2 \cdot precision \cdot recall}{precision + recall} \quad (4)$$

J. EXPLAINABLE AI (XAI)

The traditional evaluation metrics, such as accuracy, precision, recall, etc., could provide small insights into the model performance, but they didn't provide complete confidence in the model's reliability. Explainable AI is a critical aspect in understanding and learning; it refers to the tools that help to provide understandable and clear explanations for the model decision. It is using a new sophisticated approach. A well-crafted explanation can motivate individuals to learn more and delve deeper into the subject. Several famous tools used to provide explanation include LIME [67] and SHAP [68]

- The SHAP (Shapley Additive Explanations) is an effective tool to generate an understandable interpretation for the ML prediction [68]. It provides a unified framework for both feature importance at the dataset level and individual prediction at the instance level in. SHAP library is built on shapely values and games theory basis, which permit the assignment of feature contribution for each feature. SHAP library is versatile that could explain various machine learning models. SHAP works by building model g (explanation model) for the original f.

$$F(x) = g\left(X'\right) = \emptyset_0 = \sum_{i=1}^{M} \emptyset_0 X_1'^{i} \quad (5)$$

where $F(x)$ is the original model and $g\left(X'\right)$ is the explanation model $\emptyset_0 = f\,(h_x(0))$ is the model output This model could be calculated using the following equation. The shapely value is also calculated using the following equation to measure the feature contribution.

$$\emptyset_i = \sum_{s \in n \backslash i} \frac{|S|!(M - |S| - 1)!}{M!} \left[f_x(S \cup \{i\}) - f_x(S)\right] \quad (6)$$

where S set of the non-zero index in z, M is the total number of features, N all input features.

- LIME (Local interpretable model-agnostic explanation) is used to interpret the model prediction by approximating the black box with other local explanation models. LIME provides the local explanation by fitting the local model [67]. The interpretable inputs and the mapping $X = h_x\left(X'\right))$ convert the binary vector of the interpretable input of the original input. The local explanation of LIME calculated in

$$g\left(z'\right) = \emptyset_0 = \sum_{i=1}^{M} \emptyset_0 Z_i' \quad (7)$$

where $Z' \in \{0, 1\}$, M is the number of input features

V. EXPERIMENTS RESULTS

A. EXPERIMENT SETUP

Scikit-learn was employed to implement models. The dataset was divided into two parts: 70% training and 30% testing using stratified sampling methods. The training set is used to optimize and train ML models. A testing set is used to evaluate ML models based on different evaluation metrics. Stacking models compared with different ML models: RF, LR, DT, SVM. KNN, NB, KNN, and ensemble models: bagging, GradientBoosting, voting.

POS and GA were applied to select the most suitable subset features from the dataset. We used the GeneticSelectionCV library to implement GA and to implement POS. Models were applied to full features and selected features. The values of parameters for the POS algorithm are shown in Table 3. The values of parameters for the POS algorithm are shown in Table 4.

TABLE 3. Parameters of PSO.

Parameter	Value
Population size	20
Max num of generation	30
Early stopping	True
Local best weight	1
Global best weight	1
Use local random seed.	True

TABLE 4. Parameters of GA.

Parameters	Values
Crossover rate	0.5
Mutation rate	0.2
Population Size	100
Iteration number	100
population	100

B. RESULTS WITH ALL FEATURES

In this section, we explore the performance of the traditional ML, ensemble learning, and the proposed stacking models with all features.

Table 5 details the results of all ML models regarding several evaluation metrics. We can observe the following. Using traditional classifiers such as KNN with all feature sets gave a minor performance (accuracy = 76.44, precision = 76.64, recall = 76.44, and F1-score = 75.88). SVC and LR improved performance with about (1-9) % above KNN. The best performance was obtained from RF (accuracy = 91.40, precision = 91.75, recall = 91.40, and F1-score = 91.29).

From Table 5, we can see that ensemble models, including (bagging, boosting, voting, and the proposed stacking model) were enhanced performance with (1-4) above RF. Voting gave the lowest performance compared to other ensemble models (accuracy = 94.07, precision = 94.31, recall = 94.30, and F1-score = 94.30). The proposed stacking model performed best (accuracy = 96.46, precision = 96.46, recall = 96.46, and F1-score = 96.46).

TABLE 5. Results with all features.

Models	Accuracy	Precision	Recall	F1-score
RF	91.40	91.75	91.40	91.29
LR	84.47	84.42	84.47	84.42
DT	83.98	87.45	83.98	83.07
SVM	85.21	85.18	85.21	85.14
KNN	76.44	76.64	76.44	75.88
NB	89.78	90.09	89.78	89.66
Bagging	95.30	95.31	95.30	95.30
GradientBoosting	94.55	94.72	94.55	94.52
voting	94.07	94.02	94.07	94.02
Stacking	96.46	96.46	96.46	96.46

Figure 6 shows the ROC curve and AUC of models with all features. The stacking model has the highest AUC =, 96.384, while KNN has the lowest AUC at 74.366.

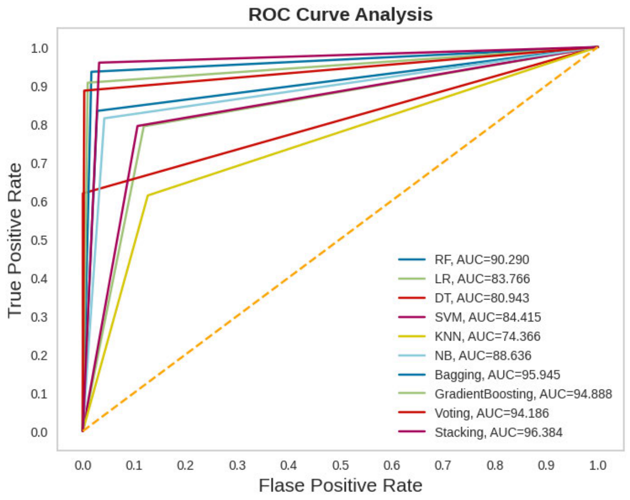

FIGURE 6. ROC with all features.

C. RESULTS WITH OPTIMIZED FEATURES USING GENETIC ALGORITHMS

In this section, we explore the performance of the traditional ML, ensemble learning, and the proposed stacking models with selected features by genetics. Table 6 details the results of all models regarding several evaluation metrics using selected features by genetics. We can observe the following. Using traditional classifiers such as KNN with all feature sets gave a minor performance (accuracy = 73.09, precision = 72.90, recall = 73.09, and F1-score = 72.71).

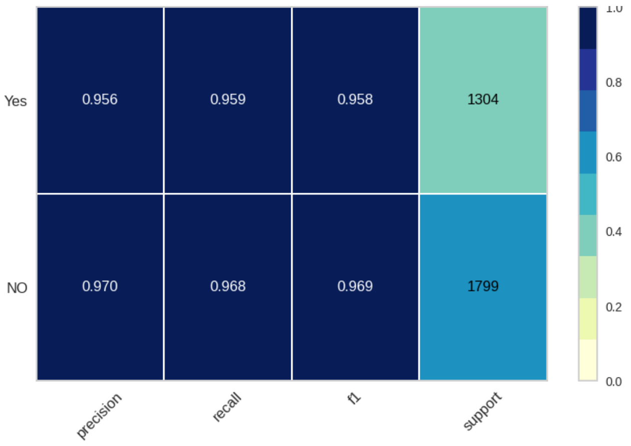

FIGURE 7. Classification report of stacking model for all features.

RF and DT improved performance with about (19) % above KNN. LR and SVM performed performance similarly. The best performance was obtained from RF (accuracy = 92.07, precision = 92.10, recall = 92.07, and F1-score = 92.04).

From Table 6, we can see that ensemble models, including (bagging, boosting, voting, and the proposed stacking model) were enhanced performance with (1-4) above RF. Voting gave the lowest performance compared to other ensemble models(accuracy = 93.33, precision = 93.63, recall = 93.33, and F1-score = 93.26). Bagging and GradientBoosting gave a similar performance. The proposed stacking model performed best (accuracy = 96.94, precision = 96.94, recall = 96.94, and F1-score = 96.94).

TABLE 6. Results of the models with selected features by genetic.

Models	Accuracy	Precision	Recall	F1-score
RF	92.07	92.10	92.07	92.04
LR	77.25	77.20	77.25	76.93
DT	91.07	91.11	91.07	91.03
SVM	78.54	78.63	78.54	78.17
KNN	73.09	72.90	73.09	72.71
NB	90.69	91.66	90.69	90.49
Bagging	95.13	95.15	95.13	95.12
GradientBoosting	94.07	94.31	94.07	94.02
voting	93.33	93.63	93.33	93.26
Stacking	96.94	96.94	96.94	96.94

Figure 8 shows the ROC curve and AUC of models with all features. The stacking model has the highest AUC = 96.779, while KNN has the lowest AUC at 71.380.

D. RESULTS WITH OPTIMIZED FEATURES USING PSO

In this section, we explore the performance of the traditional ML, ensemble learning, and the proposed stacking models with selected features by the PSO.

Table 7 details the results of all ML models regarding several evaluation metrics. We can observe the following. Using traditional classifiers such as KNN with selected feature sets gave a minor performance (accuracy = 77.83, precision = 77.99, recall = 77.83, and F1-score = 77.38).

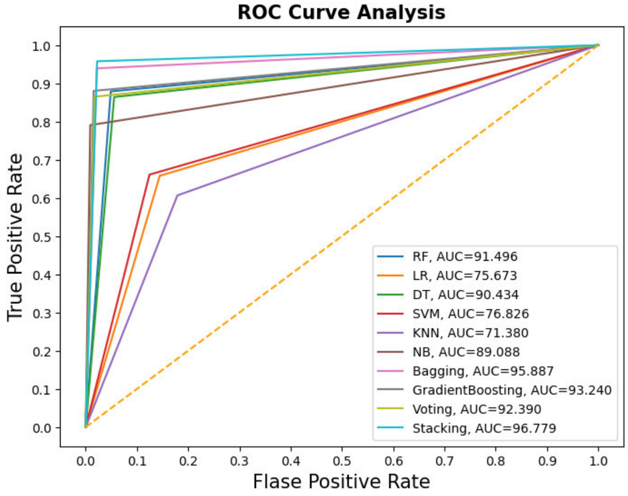

FIGURE 8. ROC curves with selected features by genetic.

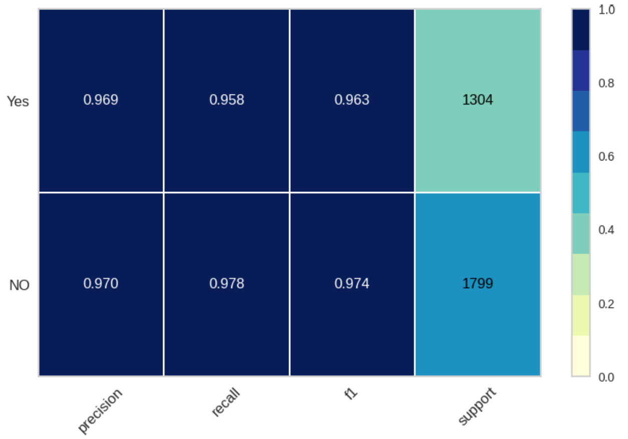

FIGURE 9. Classification report of stacking model for selected features by genetic.

RF and DT improved performance with about (19) % above KNN. SVM and LR performed performance similarly. The best performance was obtained from RF (accuracy = 93.94, precision = 93.94, recall = 93.94, and F1-score = 93.93).

From Table 7, we can see that ensemble models, including (bagging, boosting, voting, and the proposed stacking model) were enhanced performance with (1-4) above NB. voting gave the lowest performance compared to other ensemble models(accuracy = 95.10, precision = 95.28, recall = 95.10, and F1-score = 95.07). Bagging and GradientBoosting gave a similar performance. The proposed stacking model performed best (accuracy = 98.42, precision = 98.42, recall = 98.42, and F1-score = 98.42).

VI. DISCUSSION

This section presents the statistical analysis of the results, explainable artificial intelligence and compares the proposed model with the literature review.

TABLE 7. Results of the models with selected features by PSO.

Models	Accuracy	Precision	Recall	F1-score
RF	93.94	93.94	93.94	93.93
LR	82.50	82.46	82.50	82.38
DT	89.88	89.93	89.88	89.82
SVM	83.44	83.42	83.44	83.31
KNN	77.83	77.99	77.83	77.38
NB	89.78	90.05	89.78	89.67
Bagging	97.29	97.29	97.29	97.29
GradientBoosting	95.84	95.96	95.84	95.82
voting	95.10	95.28	95.10	95.07
Stacking	98.42	98.42	98.42	98.42

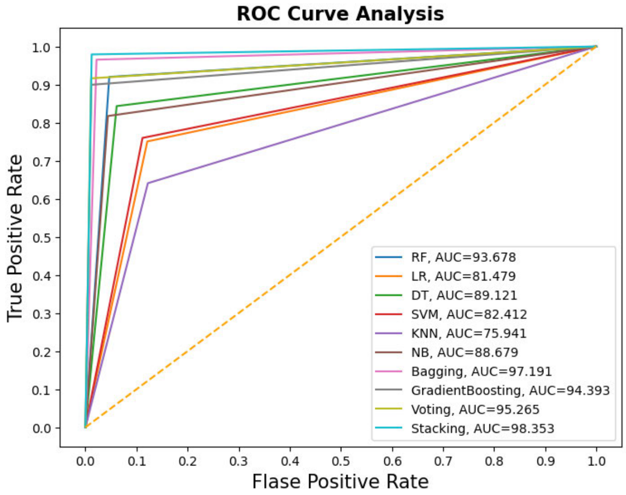

FIGURE 10. ROC curves with selected features by PSO.

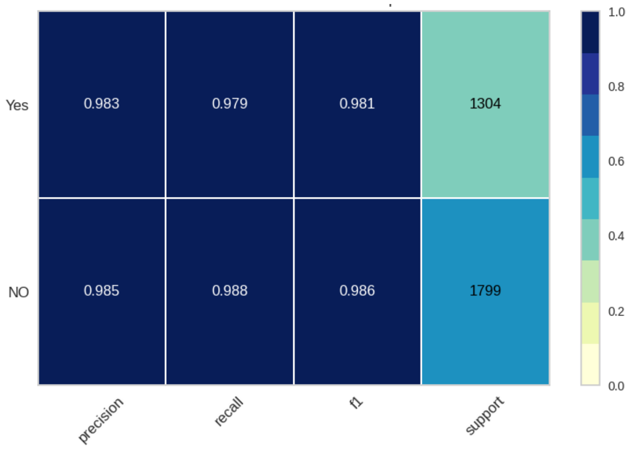

FIGURE 11. Classification report of stacking model for selected features by PSO.

A. COMPARISON BETWEEN ALL MODELS

In this study, we explore the performance of the proposed stacking model with full features and selected features. From Figure 12, we can see that the stacking model with POS has achieved the highest performance compared to full features and selected features by GA. And the stacking model with full features recorded the lowest performance.

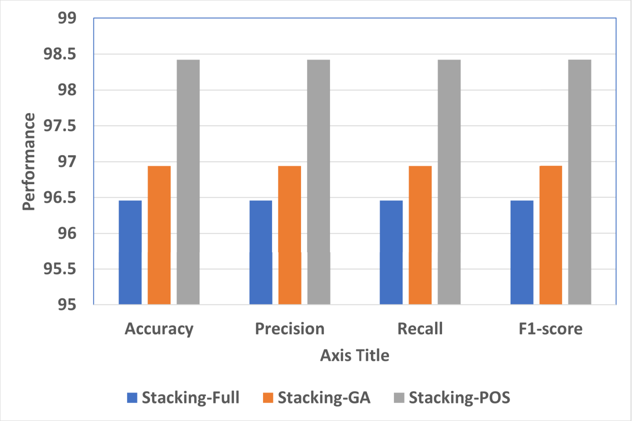

FIGURE 12. The best stacking models that were applied to full and selected features.

B. STATISTICAL ANALYSIS OF THE RESULTS

To ensure the superiority of the proposed model, we measure the statistical difference between the tested models. In this regard, we depend on Wilcoxon signed rank test [69]. It's a nonparametric test that is used to compare algorithms' performance. It works by counting the number of ties and wins. The algorithm is considered better if the number of wins exceeds the number of ties. To rank all classifiers and choose the best one. The Nemenyi test [70] was utilized to calculate the average rank for all classifiers. The rank is based on the accuracy of the classifiers. When multiple classifiers are compared with the same dataset, the results of the Nemenyi test could be visualized using the critical distance measure. Figure 13 shows the average rank of the critical difference between all traditional ensemble ML and the proposed ensemble model. After comparing all models, we conclude that the proposed ensemble model shows a significant difference between the traditional and ensemble classifiers.

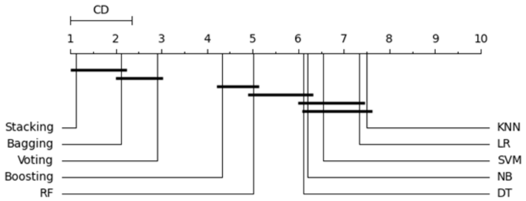

FIGURE 13. Statistical analysis of the results.

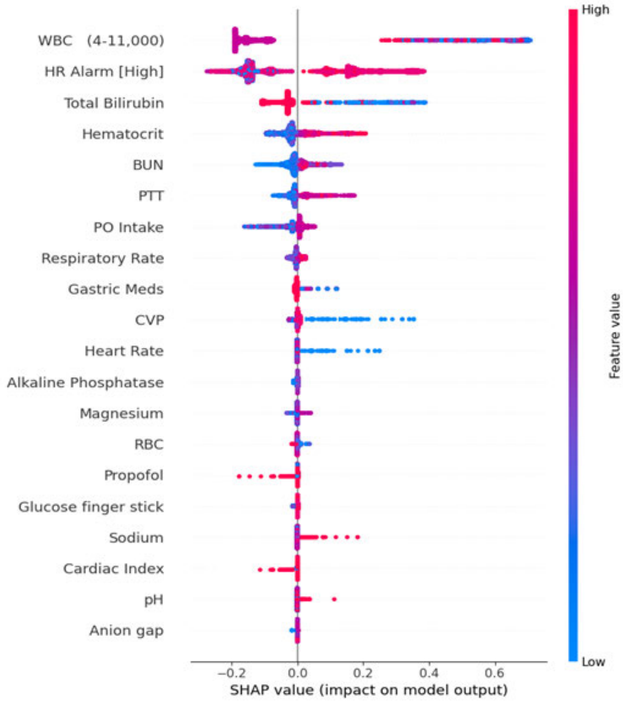

FIGURE 14. Global explanation of stacking model according to SHAP explainer.

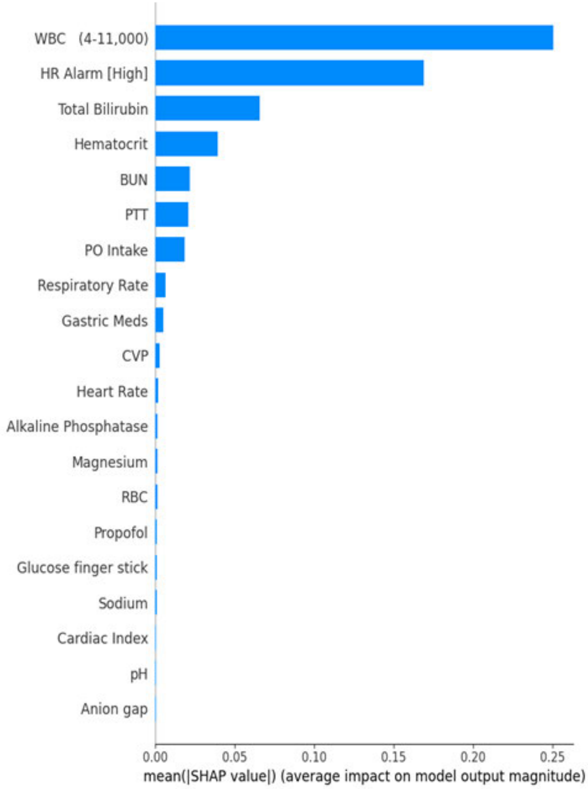

FIGURE 15. Shap Summary plot according to mean shap values.

C. EXPLAINABLE ARTIFICIAL INTELLIGENCE

Unless the promising results of the developed model, it may not be trusted from a medical expert's perspective. They need to know several things, including why the system developed that decision and which features have a high impact. Could these features be sufficient from a medical perspective? In this section, we provide an understandable explanation of the developed decision. From the experiments mentioned above, we choose to explain our proposed ensemble classifier (the most accurate classifier). We use SHAP and LIME explainers to interpret the model from Figure 14, showing the developed model's SHAP summary plot. The entire dataset offers the most significant features that significantly affect the decision, as shown in Figure 14; WBC, HR alarm, Total bilirubin, and hematocrit are considered the essential features. The horizontal line in front of each feature represents

TABLE 8. Comparing the proposed model with other models in literature.

Papers	Models	Datasets	AUC
[25]	Ensemble model	11805 patients from Brazilian university hospital.	AUC=91
[26]	Ensemble model	3326 from MMIC	AUC= 70.95
[14]	Gradient-boosted machine	24885 from US hospital	AUC=70
[27]	XGboost	from US hospital	AUC=65.91
[28]	XGBoost	5791 COVID-19 hospitalizations	AUC=91
[29]	LR	58036 patients from hospital in Asia.	AUC= 70.6
[30]	RF	1628 patients from hospital in Anhui	AUC=87
[31]	fuzzy modeling	26655 patients from MIMICII	AUC= 72
[33]	CNN-LSTM	MMICII	AUC=82.1

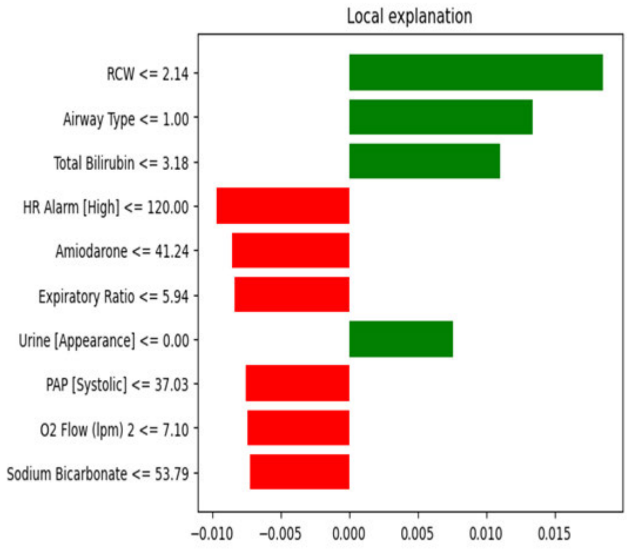

FIGURE 16. Local explanation according to LIME explainers.

the effect of features on the output, blue means it has an impact to make the model go in the lower direction, and red means it goes in the upper direction. These results are also confirmed in Figure 15, which shows the mean impact of each feature. It shows the feature names on the X-axis, and each bar's length represents the feature's importance. After leading the global explanation, we need to ensure the effectiveness of the features according to each instance. As shown in Figure 16, the LIME explainer shows the impact of each feature in a specific example. The length of each bar represents the importance of the feature. The feature highlighted in green contributes to making the model go toward class 1(readmit), while features highlighted in red make the model go toward class 0 (not readmit). The local explanation is also confirmed using the SHAP force plot, as in Figure 17 (A, B). The force plot explains in terms of a single model. Each figure shows a deep understanding case. For example, Figure 17 (A) shows how the features contributed to the model prediction according to instance 10. The model gives a predict_proba value of 0.23, and the base value of the predicted value is 0 4197: the base value is the model's average over the overall dataset. The arrows represent the value of the features (WBC = 13.8, HR ALARM = 120.1, PO intake = 137.7, hematocrit = 27.1). The big arrow refers to the features that have a significant impact, and the short arrow means the feature has a small effect. Red arrows mean that features force the model to a higher score, and blue forces the model to a lower score.

D. COMPARING THE PROPOSED MODEL WITH THE LITERATURE REVIEW

Comparing the proposed model with other models as shown in Table 8. In [25] and [26], the authors used, and AUC recorded 91 and 70.95, respectively. In [14], the authors used Gradient-boosted, and AUC recorded 70. In [27] and [28], the authors used XGboost and AUC and recorded 65.91 and 91, respectively. In [29], the authors used LR, and AUC recorded 70.6. In [30], the authors used RF, and AUC recorded 87. The authors applied fuzzy modeling in [31], and AUC recorded 72. In [33], the authors used CNN-LSTM, and AUC recorded 82.1.

VII. THREATS TO VALIDITY

To ensure the accuracy and generalizability of research findings, we should consider the potential threats to validity and take measures to minimize their impact on the study. These could be summarized in the following points.

1) Selection bias: The dataset that include in that study only includes patients from a MIMIC III dataset that aggregated from patients admitted to intensive care units (ICUs) at the Beth Israel Deaconess Medical Center in Boston, Massachusetts, USA. Therefore, we intend to test the proposed model with other real dataset to ensure the proposed model would give accurate prediction with other populations or settings.
2) Missing data: our study was built on number of vital signs and laboratory tests. Therefore, when testing the proposed model in the real-world, if certain laboratory tests or vital signs were not recorded for some patients, the model may not be able to accurately predict their risk of readmission.

Therefore, researchers need to carefully consider these factors and take steps to mitigate their effects when redesigning our model in their studies.

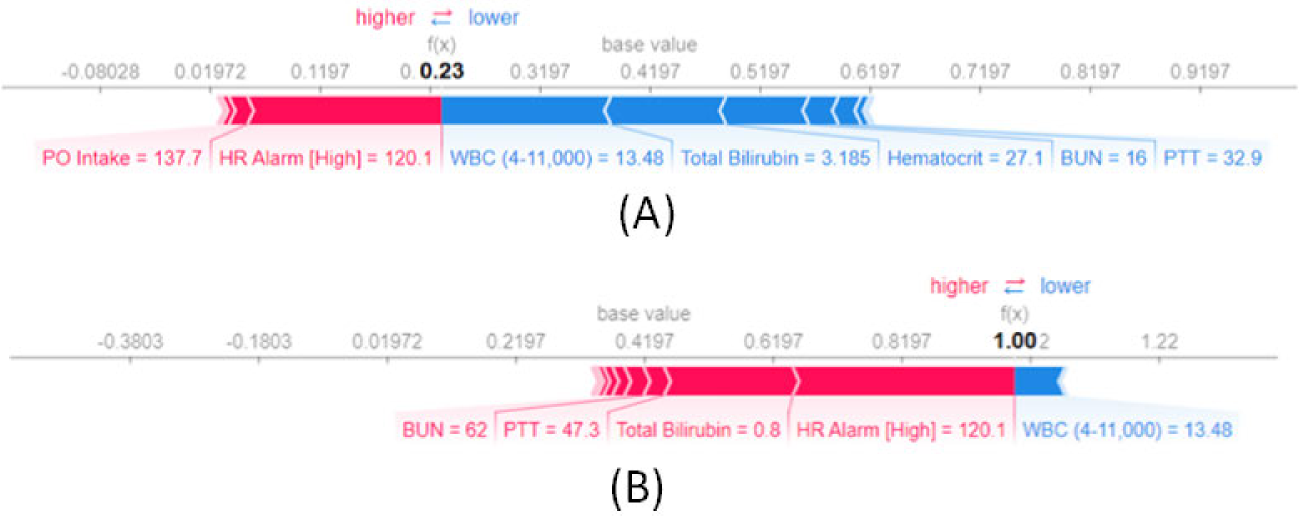

FIGURE 17. Force plot for specific instance.

VIII. CONCLUSION

Due to healthcare devices' heterogeneous and dynamic nature, real-time monitoring and timely decision-making have become crucial problems. This study introduced a clinical monitoring fog-computing-based system for remote prognosis and monitoring of the state of intensive care patients and predicted unplanned readmission outside the ICU within the next 30 days of discharge. The proposed system has three layers: data acquisition, fog computing, and a cloud server. First, a set of IoT devices called medical sensors are scattered to collect the vital signs of the patient's health conditions in real-time, in addition to the data aggregated through lab tests and medical histories. Then, data is locally processed in the fog computing layer and predicts unplanned patient readmission. The results are then sent to the cloud layer, offering sizable storage space for patient healthcare. The e-healthcare monitoring system can generate a real-time alert signal for doctors supplying e-healthcare. This proposed monitoring system used the advantages of ML approaches for developing a real-time alert signal for doctors to provide e-healthcare, accelerating decision-making, and monitoring and controlling health systems. The proposed model was trained and evaluated based on 10,644 patients extracted from the MIMIC III dataset. Feature selection methods: Genetic algorithm (GA) and practical swarm optimization (PSO) are used to choose the optimal feature subset from detests. Moreover, Different traditional ML models, ensemble learning models, and the proposed stacking models are applied to full features and selected features to predict readmission after 30 days of ICU discharge. The study results showed that the proposed stacking model with features chosen by PSO outperforms other traditional and ensemble ML, with an accuracy of 98.42, a precision of 98.42, a recall of 98.42, and an F1-Score of 98.42. In addition, we provide local and global explanations of the developed model to ensure its efficiency, effectiveness, and trustworthiness. The development of interpretable models such as the proposed model is necessary to improve the generalization ability of the proposed model and enhance the integration of the predictive model in the clinical process. Future studies will concentrate on ensuring the generalization ability of the proposed model by validating it using various datasets and different diseases. They will also use unstructured data from MIMIC datasets such as clinical notes, images, etc. To gain valuable information and improve the model's overall performance. In our future work, we plan to aggregate more data to ensure the model's generalizability. Additionally, we intend to analyze the model from a computational complexity perspective to ensure efficiency. Furthermore, we plan to deploy the developed model in a real clinical system to evaluate its performance in a practical setting.

DATA AVAILABILITY

All datasets used to support the findings of this study are available from the direct link in the dataset citations.

CONFLICT OF INTERESTS

All authors declare that they have no conflicts of interest.

ACKNOWLEDGMENT

This research was supported by the Basic Science Research Program through a National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (NRF-2021R1A2B5B02002599). Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2023R407), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

REFERENCES

[1] C. K. McIlvennan, Z. J. Eapen, and L. A. Allen, "Hospital readmissions reduction program," *Circulation*, vol. 131, no. 20, pp. 1796–1803, 2015.

[2] L. Huang, A. L. Shea, H. Qian, A. Masurkar, H. Deng, and D. Liu, ''Patient clustering improves efficiency of federated machine learning to predict mortality and hospital stay time using distributed electronic medical records,'' *J. Biomed. Informat.*, vol. 99, Nov. 2019, Art. no. 103291.

[3] R. S. Anand, P. Stey, S. Jain, D. R. Biron, H. Bhatt, K. Monteiro, E. Feller, M. L. Ranney, I. N. Sarkar, and E. S. Chen, ''Predicting mortality in diabetic ICU patients using machine learning and severity indices,'' *AMIA Summits Transl. Sci. Proc.*, vol. 2018, p. 310, Jan. 2018.

[4] M. Scherpf, F. Gräßer, H. Malberg, and S. Zaunseder, ''Predicting sepsis with a recurrent neural network using the MIMIC III database,'' *Comput. Biol. Med.*, vol. 113, Oct. 2019, Art. no. 103395.

[5] S. Layeghian Javan, M. M. Sepehri, M. L. Javan, and T. Khatibi, ''An intelligent warning model for early prediction of cardiac arrest in sepsis patients,'' *Comput. Methods Programs Biomed.*, vol. 178, pp. 47–58, Sep. 2019.

[6] Z. Xu, Y. Feng, Y. Li, A. Srivastava, P. Adekkanattu, J. S. Ancker, G. Jiang, R. C. Kiefer, K. Lee, J. A. Pacheco, and L. V. Rasmussen, ''Predictive modeling of the risk of acute kidney injury in critical care: A systematic investigation of the class imbalance problem,'' *AMIA Summits Transl. Sci. Proc.*, vol. 2019, p. 809, May 2019.

[7] S. Barbieri, J. Kemp, O. Perez-Concha, S. Kotwal, M. Gallagher, A. Ritchie, and L. Jorm, ''Benchmarking deep learning architectures for predicting readmission to the ICU and describing patients-at-risk,'' *Sci. Rep.*, vol. 10, no. 1, p. 1111, Jan. 2020.

[8] O. Gajic, M. Malinchoc, T. B. Comfere, M. R. Harris, A. Achouiti, M. Yilmaz, M. J. Schultz, R. D. Hubmayr, B. Afessa, and J. C. Farmer, ''The stability and workload index for transfer score predicts unplanned intensive care unit patient readmission: Initial development and validation,'' *Crit. Care Med.*, vol. 36, no. 3, pp. 676–682, Mar. 2008.

[9] F. S. Hosein, N. Bobrovitz, S. Berthelot, D. Zygun, W. A. Ghali, and H. T. Stelfox, ''A systematic review of tools for predicting severe adverse events following patient discharge from intensive care units,'' *Crit. Care*, vol. 17, pp. 1–10, Jun. 2013.

[10] K. Reini, M. Fredrikson, and A. Oscarsson, ''The prognostic value of the modified early warning score in critically ill patients: A prospective, observational study,'' *Eur. J. Anaesthesiol.*, vol. 29, no. 3, pp. 152–157, Mar. 2012.

[11] O. Badawi and M. J. Breslow, ''Readmissions and death after ICU discharge: Development and validation of two predictive models,'' *PLoS ONE*, vol. 7, no. 11, Nov. 2012, Art. no. e48758.

[12] I. Ouanes, C. Schwebel, A. Français, C. Bruel, F. Philippart, A. Vesin, L. Soufir, C. Adrie, M. Garrouste-Orgeas, J.-F. Timsit, and B. Misset, ''A model to predict short-term death or readmission after intensive care unit discharge,'' *J. Crit. Care*, vol. 27, no. 4, p. 422, Aug. 2012.

[13] S. A. Frost, V. Tam, E. Alexandrou, L. Hunt, Y. Salamonson, P. M. Davidson, M. J. Parr, and K. M. Hillman, ''Readmission to intensive care: Development of a nomogram for individualising risk,'' *Crit. Care Resuscitation*, vol. 12, no. 2, pp. 83–89, 2010.

[14] J. C. Rojas, K. A. Carey, D. P. Edelson, L. R. Venable, M. D. Howell, and M. M. Churpek, ''Predicting intensive care unit readmission with machine learning using electronic health record data,'' *Ann. Amer. Thoracic Soc.*, vol. 15, no. 7, pp. 846–853, Jul. 2018.

[15] M. Pishgar, J. Theis, M. Del Rios, A. Ardati, H. Anahideh, and H. Darabi, ''Prediction of unplanned 30-day readmission for ICU patients with heart failure,'' *BMC Med. Informat. Decis. Making*, vol. 22, no. 1, pp. 1–12, Dec. 2022.

[16] D.-P. Nguyen, N. Paris, and A. Parrot, ''Accurate and reproducible prediction of icu readmissions,'' *MedRxiv*, 2019.

[17] N. El-Rashidy, A. Sedik, A. I. Siam, and Z. H. Ali, ''An efficient edge/cloud medical system for rapid detection of level of consciousness in emergency medicine based on explainable machine learning models,'' *Neural Comput. Appl.*, vol. 35, no. 14, pp. 10695–10716, May 2023.

[18] D. Gowda V, A. Sharma, B. K. Rao, R. Shankar, P. Sarma, A. Chaturvedi, and N. Hussain, ''Industrial quality healthcare services using Internet of Things and fog computing approach,'' *Meas., Sensors*, vol. 24, Dec. 2022, Art. no. 100517.

[19] J. Kharel, H. T. Reda, and S. Y. Shin, ''Fog computing-based smart health monitoring system deploying LoRa wireless communication,'' *IETE Tech. Rev.*, vol. 36, no. 1, pp. 69–82, Jan. 2019.

[20] F. M. Talaat, ''Effective prediction and resource allocation method (EPRAM) in fog computing environment for smart healthcare system,'' *Multimedia Tools Appl.*, vol. 81, no. 6, pp. 8235–8258, Mar. 2022.

[21] Z. H. Ali and H. A. Ali, ''EEOMA: End-to-end oriented management architecture for 6G-enabled drone communications,'' *Peer-Peer Netw. Appl.*, vol. 15, no. 2, pp. 1232–1254, Mar. 2022.

[22] L. Turgeman and J. H. May, ''A mixed-ensemble model for hospital readmission,'' *Artif. Intell. Med.*, vol. 72, pp. 72–82, Sep. 2016.

[23] J. Futoma, J. Morris, and J. Lucas, ''A comparison of models for predicting early hospital readmissions,'' *J. Biomed. Informat.*, vol. 56, pp. 229–238, Aug. 2015.

[24] C. Hebert, C. Shivade, R. Foraker, J. Wasserman, C. Roth, H. Mekhjian, S. Lemeshow, and P. Embi, ''Diagnosis-specific readmission risk prediction using electronic health data: A retrospective cohort study,'' *BMC Med. Informat. Decis. Making*, vol. 14, no. 1, pp. 1–9, Dec. 2014.

[25] M. Loreto, T. Lisboa, and V. P. Moreira, ''Early prediction of ICU readmissions using classification algorithms,'' *Comput. Biol. Med.*, vol. 118, Mar. 2020, Art. no. 103636.

[26] T. Desautels, R. Das, J. Calvert, M. Trivedi, C. Summers, D. J. Wales, and A. Ercole, ''Prediction of early unplanned intensive care unit readmission in a UK tertiary care hospital: A cross-sectional machine learning approach,'' *BMJ Open*, vol. 7, no. 9, Sep. 2017, Art. no. e017199.

[27] Y. Huang, A. Talwar, Y. Lin, and R. R. Aparasu, ''Machine learning methods to predict 30-day hospital readmission outcome among US adults with pneumonia: Analysis of the national readmission database,'' *BMC Med. Informat. Decis. Making*, vol. 22, no. 1, pp. 1–14, Nov. 2022.

[28] M. R. Afrash, H. Kazemi-Arpanahi, M. Shanbehzadeh, R. Nopour, and E. Mirbagheri, ''Predicting hospital readmission risk in patients with COVID-19: A machine learning approach,'' *Informat. Med. Unlocked*, vol. 30, 2022, Art. no. 100908.

[29] X. Gao, S. Alam, P. Shi, F. Dexter, and N. Kong, ''Interpretable machine learning models for hospital readmission prediction: A two-step extracted regression tree approach,'' *BMC Med. Informat. Decis. Making*, vol. 23, no. 1, pp. 1–11, Jun. 2023.

[30] X. Li, Y. Zhu, W. Zhao, R. Shi, Z. Wang, H. Pan, and D. Wang, ''Machine learning algorithm to predict the in-hospital mortality in critically ill patients with chronic kidney disease,'' *Renal Failure*, vol. 45, no. 1, Dec. 2023, Art. no. 2212790.

[31] A. S. Fialho, F. Cismondi, S. M. Vieira, S. R. Reti, J. M. C. Sousa, and S. N. Finkelstein, ''Data mining using clinical physiology at discharge to predict ICU readmissions,'' *Exp. Syst. Appl.*, vol. 39, no. 18, pp. 13158–13165, Dec. 2012.

[32] Z. Li, X. Xing, B. Lu, and Z. Li, ''Early prediction of 30-day ICU readmissions using natural language processing and machine learning,'' 2019, *arXiv:1910.02545*.

[33] T. Zebin and T. J. Chaussalet, ''Design and implementation of a deep recurrent model for prediction of readmission in urgent care using electronic health records,'' in *Proc. IEEE Conf. Comput. Intell. Bioinf. Comput. Biol. (CIBCB)*, Jul. 2019, pp. 1–5.

[34] Y.-W. Lin, Y. Zhou, F. Faghri, M. J. Shaw, and R. H. Campbell, ''Analysis and prediction of unplanned intensive care unit readmission using recurrent neural networks with long short-term memory,'' *PLoS ONE*, vol. 14, no. 7, Jul. 2019, Art. no. e0218942.

[35] X. Liu, Y. Chen, J. Bae, H. Li, J. Johnston, and T. Sanger, ''Predicting heart failure readmission from clinical notes using deep learning,'' in *Proc. IEEE Int. Conf. Bioinf. Biomed. (BIBM)*, Nov. 2019, pp. 2642–2648.

[36] A. E. W. Johnson, T. J. Pollard, L. Shen, L.-W.-H. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. A. Celi, and R. G. Mark, ''MIMIC-III, a freely accessible critical care database,'' *Sci. Data*, vol. 3, no. 1, pp. 1–9, May 2016.

[37] R. Yu, Y. Zheng, R. Zhang, Y. Jiang, and C. C. Y. Poon, ''Using a multitask recurrent neural network with attention mechanisms to predict hospital mortality of patients,'' *IEEE J. Biomed. Health Informat.*, vol. 24, no. 2, pp. 486–492, Feb. 2020.

[38] R. Sadeghi, T. Banerjee, and W. Romine, ''Early hospital mortality prediction using vital signals,'' *Smart Health*, vols. 9–10, pp. 265–274, Dec. 2018.

[39] S. Venkatesh and S. Gopal, ''Robust heteroscedastic probabilistic neural network for multiple source partial discharge pattern recognition—Significance of outliers on classification capability,'' *Exp. Syst. Appl.*, vol. 38, no. 9, pp. 11501–11514, Sep. 2011.

[40] L. Greco, G. Luta, M. Krzywinski, and N. Altman, ''Analyzing outliers: Robust methods to the rescue,'' *Nature Methods*, vol. 16, no. 4, pp. 275–277, 2019.

[41] T. K. Moon, ''The expectation-maximization algorithm,'' *IEEE Signal Process. Mag.*, vol. 13, no. 6, pp. 47–60, Nov. 1996.

[42] D. Joenssen and U. Bankhofer, “Hot deck methods for imputing missing data hot deck methods for imputing missing data the effects of limiting donor usage,” Tech. Rep., 2012.

[43] E. Caballero-Ruiz, G. García-Sáez, M. Rigla, M. Villaplana, B. Pons, and M. E. Hernando, “A web-based clinical decision support system for gestational diabetes: Automatic diet prescription and detection of insulin needs,” *Int. J. Med. Informat.*, vol. 102, pp. 35–49, Jun. 2017.

[44] H. Garg, “A hybrid PSO-GA algorithm for constrained optimization problems,” *Appl. Math. Comput.*, vol. 274, pp. 292–305, Feb. 2016.

[45] K. Premalatha and A. Natarajan, “Hybrid PSO and GA for global maximization,” *Int. J. Open Problems Compt. Math.*, vol. 2, no. 4, pp. 597–608, 2009.

[46] X.-S. Yang, *Nature-Inspired Optimization Algorithms*. Cambridge, MA, USA: Academic Press, 2020.

[47] A. G. Gad, “Particle swarm optimization algorithm and its applications: A systematic review,” *Arch. Comput. Methods Eng.*, vol. 29, no. 5, pp. 2531–2561, Aug. 2022.

[48] S. B. Sakri, N. B. A. Rashid, and Z. M. Zain, “Particle swarm optimization feature selection for breast cancer recurrence prediction,” *IEEE Access*, vol. 6, pp. 29637–29647, 2018.

[49] B. Esme, B. Sankur, and E. Anarvm, “Facial feature extraction using genetic algorithms,” in *Proc. 8th Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 1996, pp. 1–4.

[50] M. Pei, E. D. Goodman, W. F. Punch, and Y. Ding, “Genetic algorithms for classification and feature extraction,” in *Proc. Classification Soc. Conf.*, 1995, pp. 1–28.

[51] S. Fong, S. Deb, X.-S. Yang, and Y. Zhuang, “Towards enhancement of performance of K-means clustering using nature-inspired optimization algorithms,” *Sci. World J.*, vol. 2014, pp. 1–16, Jan. 2014.

[52] Y. Liu, Y. Wang, and J. Zhang, “New machine learning algorithm: Random forest,” in *Proc. Inf. Comput. Appl., 3rd Int. Conf.* Chengde, China: Springer, 2012, pp. 246–252.

[53] M. R. Segal, “Machine learning benchmarks and random forest regression,” Tech. Rep., 2004.

[54] G. Tripepi, K. Jager, F. Dekker, and C. Zoccali, “Linear and logistic regression analysis,” *Kidney Int.*, vol. 73, no. 7, pp. 806–810, 2008.

[55] J. J. Guido, P. C. Winters, and A. B. Rains, “Logistic regression basics,” M.S. thesis, Univ. Rochester Med. Center, Rochester, NY, USA, 2006.

[56] A. Navada, A. N. Ansari, S. Patil, and B. A. Sonkamble, “Overview of use of decision tree algorithms in machine learning,” in *Proc. IEEE Control Syst. Graduate Res. Colloq.*, Jun. 2011, pp. 37–42.

[57] B. Charbuty and A. Abdulazeez, “Classification based on decision tree algorithm for machine learning,” *J. Appl. Sci. Technol. Trends*, vol. 2, no. 1, pp. 20–28, Mar. 2021.

[58] M. Somvanshi, P. Chavan, S. Tambade, and S. V. Shinde, “A review of machine learning techniques using decision tree and support vector machine,” in *Proc. Int. Conf. Comput. Commun. Control Autom. (ICCUBEA)*, Aug. 2016, pp. 1–7.

[59] H. H. Patel and P. Prajapati, “Study and analysis of decision tree based classification algorithms,” *Int. J. Comput. Sci. Eng.*, vol. 6, no. 10, pp. 74–78, Oct. 2018.

[60] L. E. Peterson, “K-nearest neighbor,” *Scholarpedia*, vol. 4, no. 2, p. 1883, 2009.

[61] G. I. Webb, E. Keogh, and R. Miikkulainen, “Naïve Bayes,” *Encyclopedia Mach. Learn.*, vol. 15, no. 1, pp. 713–714, 2010.

[62] E. Frank, L. Trigg, G. Holmes, and I. H. Witten, “Naïve Bayes for regression,” *Mach. Learn.*, vol. 41, pp. 5–25, Oct. 2000.

[63] N. El-Rashidy, S. El-Sappagh, T. Abuhmed, S. Abdelrazek, and H. M. El-Bakry, “Intensive care unit mortality prediction: An improved patient-specific stacking ensemble model,” *IEEE Access*, vol. 8, pp. 133541–133564, 2020.

[64] N. El-Rashidy, T. Abuhmed, L. Alarabi, H. M. El-Bakry, S. Abdelrazek, F. Ali, and S. El-Sappagh, “Sepsis prediction in intensive care unit based on genetic feature optimization and stacked deep ensemble learning,” *Neural Comput. Appl.*, vol. 54, pp. 3603–3632, Nov. 2021.

[65] M. A. Ganaie, M. Hu, A. Malik, M. Tanveer, and P. Suganthan, “Ensemble deep learning: A review,” *Eng. Appl. Artif. Intell.*, vol. 115, Oct. 2022, Art. no. 105151.

[66] A. Almulihi, H. Saleh, A. M. Hussien, S. Mostafa, S. El-Sappagh, K. Alnowaiser, A. A. Ali, and M. Refaat Hassan, “Ensemble learning based on hybrid deep learning model for heart disease early prediction,” *Diagnostics*, vol. 12, no. 12, p. 3215, Dec. 2022.

[67] M. T. Ribeiro, S. Singh, and C. Guestrin, “‘Why should I trust you?’ Explaining the predictions of any classifier,” in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2016, pp. 1135–1144.

[68] S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–10.

[69] J. Demšar, “Statistical comparisons of classifiers over multiple data sets,” *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, Jan. 2006.

[70] S. Garcia and F. Herrera, “An extension on ‘statistical comparisons of classifiers over multiple data sets’ for all pairwise comparisons,” *J. Mach. Learn. Res.*, vol. 9, no. 12, pp. 1–18, 2008.

MAALI ALABDULHAFITH was born in September 1985, in Saudi Arabia. She received the Ph.D. degree in computer science from Dalhousie University, Halifax, Canada, in 2018. In 2014, she joined the College of Computer and Information Science (CCIS), Princess Noura University (PNU), as a Lecturer and was promoted to Assistant Professor, in 2018. She is currently the Director of Data Management and Performance Measurement at CCIS, PNU, overlooking and managing the strategy of the college. Her research interests lie in the areas of machine learning, data analytics, emerging wireless technology, and technology applications in health care.

HAGER SALEH received the Ph.D. degree from the Faculty of Computers and Information, Minia University, Egypt. She is an Assistant Professor with the Faculty of Computers and Artificial Intelligence, South Valley University, Hurghada, Egypt. She is a Researcher Member of the Big Data Team, Egypt. Her research interests are centered on big data analytics, data mining, sentiment analysis, natural language processing, machine learning, deep learning, and streaming data.

HELA ELMANNAI received the Ph.D. degree in information technology from SUPCOM, Tunisia. She is an Assistant Professor with the Department of Information Technology, College of Computer and Information Sciences, Princess Nourah Bint Abdulrahman University, Saudi Arabia. Her research interests include artificial intelligence, networking, blockchain, and engineering applications.

ZAINAB HASSAN ALI received the B.Sc. degree in computer science and engineering from the Faculty of Electronic Engineering, Menofia University, in 2009, the master’s degree in telecommunication and computer networks, in 2016, and the Ph.D. degree from the Department of Computer Engineering and Control Systems, Faculty of Engineering, Mansoura University, in 2020. She is an Associate Professor with the Embedded Network Systems and Technology Department, Faculty of Artificial Intelligence, Kafrelsheikh University, Kafr El-Sheikh, Egypt. She was an Associate Professor with the Faculty of Graduate Studies for Statistical Research, Cairo University, from 2020 to 2022. She was an Instructor with Huawei Company, from 2019 to 2023. She has received a lot of badges and certificates from Huawei, Cisco, and IBM in the fields of the Internet of Things, artificial intelligence, cloud computing, and predictive analysis. She served as a Reviewer for many high-quality journals, including *Journal of Artificial Intelligence*, *The Computer Journal* (Oxford Academic), and *Journal of Supercomputing* (Springer).

SHAKER EL-SAPPAGH received the bachelor's and master's degrees in computer science from the Department of Information Systems, Faculty of Computers and Information, Cairo University, Egypt, in 1997 and 2007, respectively, and the Ph.D. degree in computer science from the Department of Information Systems, Faculty of Computers and Information, Mansoura University, Mansoura, Egypt, in 2015. In 2003, he joined as a Teaching Assistant with the Department of Information Systems, Faculty of Computers and Information, Minia University, Egypt. Since June 2016, he has been an Assistant Professor with the Department of Information Systems, Faculty of Computers and Information, Benha University. From 2018 to 2020, he was a Research Professor with the UWB Wireless Communications Research Center, Department of Information and Communication Engineering, Inha University, South Korea. He was a Research Professor at the Centro Singular de Investigaci on en Tecnolox as Intelixentes (CiTIUS), Universidade de Santiago de Compostela, Santiago de Compostela, Spain, in 2021. He is currently an Associate Professor with Galala University, Egypt. He is also a Senior Researcher with the College of Computing and Informatics, Sungkyunkwan University, South Korea. His publications in clinical decision support systems and semantic intelligence. His current research interests include machine learning, medical informatics, (fuzzy) ontology engineering, distributed and hybrid clinical decision support systems, semantic data modeling, fuzzy expert systems, and cloud computing. He is a reviewer of many journals and he is very interested in the diseases' diagnosis and treatment research.

JONG-WAN HU received the M.S. degree from the G.W.W. School of Mechanical Engineering and the School of Civil and Environmental Engineering, Georgia Institute of Technology, and the Ph.D. degree from the Georgia Institute of Technology. He is currently an Assistant Professor with the University of Incheon. His research interests are in the area of computational solid mechanics, composite materials, and plasticity modeling.

NORA EL-RASHIDY received the M.S. and Ph.D. degrees from the Faculty of Computer Science and Information, Mansoura University, Egypt, in 2016 and 2020, respectively. She is an Associate Professor with the Machine Learning and Information Retrieval Department, Faculty of Artificial Intelligence, Kafrelsheikh University. She is an Assistant Professor with Galala University and an Assistant Professor with the Machine Learning and Information Retrieval Department, Kafr Elsheikh University, where she is the Manager of the Measurement and Assessment Central Unit. She is also a free Lancer Net and Share Point Developer and an ASP Net Developer to finally have ten years of successful professional work experience. Her research interests include machine learning, medical informatics, distributed and hybrid clinical decision support systems, big data, and cloud computing. She is a reviewer in many journals, and she is very interested in disease diagnosis and treatment research.