Neural Computing and Applications
https://doi.org/10.1007/s00521-021-06631-1

ORIGINAL ARTICLE

Sepsis prediction in intensive care unit based on genetic feature optimization and stacked deep ensemble learning

Nora El-Rashidy[1] · Tamer Abuhmed[2] · Louai Alarabi[3] · Hazem M. El-Bakry[4] · Samir Abdelrazek[4] · Farman Ali[5] · Shaker El-Sappagh[6]

Received: 7 May 2021 / Accepted: 6 October 2021

Abstract
Sepsis is a life-threatening disease that is associated with organ dysfunction. It occurs due to the body's dysregulated response to infection. It is difficult to identify sepsis in its early stages, this delay in identification has a dramatic effect on mortality rate. Developing prognostic tools for sepsis prediction has been the focus of various studies over previous decades. However, most of these studies relied on tracking a limited number of features, as such, these approaches may not predict sepsis sufficiently accurately in many cases. Therefore, in this study, we concentrate on building a more accurate and medically relevant predictive model for identifying sepsis. First, both NSGA-II (a multi-objective genetic algorithm optimization approach) and artificial neural networks are used concurrently to extract the optimal feature subset from patient data. In the next stage, a deep learning model is built based on the selected optimal feature set. The proposed model has two layers. The first is a deep learning classification model used to predict sepsis. This is a stacking ensemble of neural network models that predicts which patients will develop sepsis. For patients who were predicted to have sepsis, data from their first six hours after admission to the ICU are retrieved, this data is then used for further model optimization. Optimization based on this small, recent timeframe leads to an increase in the effectiveness of our classification model compared to other models from previous works. In the second layer of our model, a multitask regression deep learning model is used to identify the onset time of sepsis and the blood pressure at that time in patients that were predicted to have sepsis by the first layer. Our study was performed using the medical information from the intensive care MIMIC III real-world dataset. The proposed classification model achieved 0.913, 0.921, 0.832, 0.906 for accuracy, specificity, sensitivity, and AUC, respectively. In addition, the multitask regression model obtained an RMSE of 10.26 and 9.22 for predicting the onset time of sepsis and the blood pressure at that time, respectively. There are no other studies in the literature that can accurately predict the status of sepsis in terms of its onset time and predict medically verifiable quantities like blood pressure to build confidence in the onset time prediction. The proposed model is medically intuitive and achieves superior performance when compared to all other current state-of-the-art approaches.

Keywords Ensemble classifier · Deep learning · Feature optimization · Multitask learning · Sepsis prediction

1 Introduction

Sepsis is an immune-mediated response to organ dysfunction infections, the condition is life-threatening, prevalent, and costly. The incidence of sepsis is increasing by approximately 13% per year [1]. Sepsis is classified as one of the leading causes of in-hospital mortality and leads to increased risks of cognitive impairment and permanent organ damage for surviving patients. Its diagnosis is a challenge for physicians due to its multifactorial characteristics. The first definition of sepsis was developed in 1991, and it works by defining a practical framework for systemic inflammatory response syndrome (SIRS), which classifies sepsis into three different levels: sepsis, severe sepsis, and septic shock [2]. In 2001, this definition was expanded by adding another list of vital signs to the SIRS criteria to better detect sepsis [3]. In 2016, sepsis was redefined (and is now known as sepsis 3) to facilitate better sepsis detection processes [4]. However, the sepsis 3 definition has led to various potential problems in its diagnosis

Extended author information available on the last page of the article

Published online: 03 November 2021

and detection procedures due to the downgrading of detection causing a higher mortality rate. Therefore, in consideration of the problems with the sepsis 3 definition, in this work, we decided to follow the initial definition. Sepsis 1 is identified based on the existence of two criteria at the same time: SIRS and suspected infection.

Sepsis is considered the leading cause of hospital mortality for ICU patients with a mortality rate of over 45% [5], 6]. Moreover, sepsis carries a high associated risk of cardiac arrest [7]. The major tent poles of sepsis treatment are early diagnosis and the rapid initiation of treatment. Several studies have demonstrated that early detection and treatment contribute to reductions in the mortality rate and in medical expenditure [8]. The authors in [9] found that sepsis patients had a survival rate of up to 80% if treatment is received within the first hour of diagnosis, each hour of delay in treatment was found to increase the mortality rate by 8%. Another study [10] has shown that the survival probability among sepsis patients is highly dependent on the timing of the antibiotic's intervention. Most studies concentrated on predicting sepsis have depended on a small number of features to achieve this, such as in [11–13]. Regardless of the good results achieved in some studies, depending on such small numbers of features is suboptimal when discriminating sepsis cases from other diagnoses. There are many reasons for this, including: (1) the definition of organ dysfunction in relation to sepsis can be unclear as it may occur for reasons other than sepsis [14]; (2) requiring the presence of infection before allowing a sepsis prediction makes it difficult to identify sepsis when the infection is not certain, several studies have reported that organ dysfunction that returns to clinical conditions is commonly observed and this occurrence is considered difficult to distinguish from simultaneous infections; (3) the new definition of sepsis considers it to be a syndrome, this means we should treat all diseases present with similar diagnostic processes. This approach may not be suitable for patients with specific conditions such as cancer or chronic heart disease [15]. However, none of the previous literature has focused on investigating the reasons sepsis develops in the first place, which means their outputs are not accepted clinically. Therefore, this study mainly focuses on both predicting sepsis and clarifying the cause of sepsis to help define suitable sepsis treatment in addition to ensuring timely treatment in cases when sepsis is identified. Alongside predicting sepsis, we also predict its onset time and the patient's blood pressure at this time.

Multitask learning (MTL) is a technique where several associated tasks are optimized concurrently, to achieve this the learning parameters are partially shared [16,17]. By exchanging features between related tasks, MTL can enhance model generalization by exploiting hidden information among related tasks. Additionally, MTL creates more stable models because linking multiple tasks works as a regularization feature in the resulting model. MTL approaches vary in terms of model structure, information sharing level, and optimization technique used [18], as shown in Fig. 1. This study has a two-stage approach to the problem at hand. In the first stage, we predict the occurrence of sepsis in ICU patients. In the second stage, MTL is used to simultaneously predict the onset time of the disease and the patient's blood pressure at that time. The predicted blood pressure can be used as a further data point that is straightforward to verify and gives clinical credibility to the model's prediction of sepsis onset time. This information is helpful to physicians and increases their trust in the model's decisions.

An artificial neural network is a series of algorithms that aims to identify underlying relationships within a set of data through a set of processes the system is put through while being trained [19]. Although a multilayer ANN could theoretically approximate any nonlinear function, its application always brings several challenges due to the high dimensional data that is used with these models. Therefore, a preprocessing step is typically needed prior to data fitting [20]. Feature subset selection using evolutionary algorithms is considered a promising technique [21]. Several works have been developed techniques to achieve this [22, 23]. In this paper, we use a popular multi-objective feature selection technique known as the non-dominated sorting genetic technique II (NSGA II) [24]. This methodology selects the smallest number of features that can provide the best performance and has achieved superior performance to other approaches in the literature. NSGA-II works based on the concept of non-dominated sorting and crowding distance. It ranks the features according to feature importance to get the optimal number of the most important features. Its selection process is carried out through two main tasks: first, we minimize the number of features in use. Second, we calculate the classification error using a 1-NN classifier to evaluate classifier performance and compute the classification error. These steps are repeated until we reach the minimal set of features that gives the smallest error. In addition to the challenges posed by preprocessing, training the ANN itself is also considered a taxing challenge [25]. This is because the training process cannot guarantee optimal ANN weights, which leaves a decent chance of ending up with a high variance model. This challenge can be overcome, and state-of-the-art performance achieved by combining the outputs of various diverse ANN models, this process is known as ensemble learning [26]. Various studies have demonstrated that a good ensemble is one where all the ensemble's sub models are both accurate and independent in terms of their model errors [27, 28].

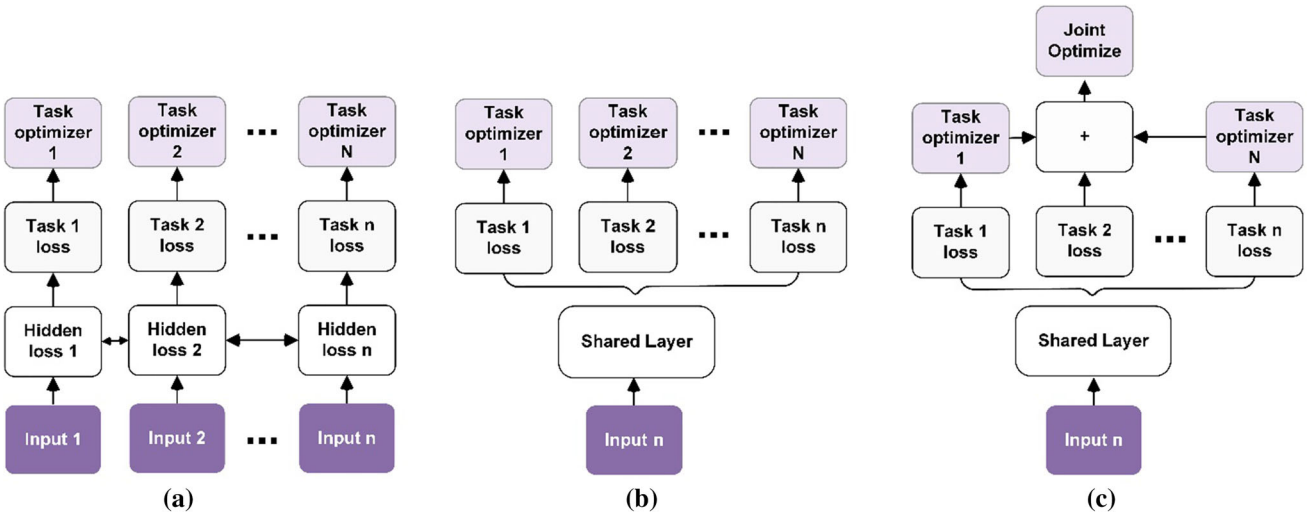

Fig. 1 Multitask architecture models: (**a**) Soft training: where each task has its own parameters and model, this is distance regularized to encourage distances to be similar, (**b**) Alternate training: this model allows sharing of information between tasks, (**c**) Joint training: this model allows different parts of the model to share parts of their structure in addition to data statistics

The main objective of this study is to highlight the importance of using all the various vital signs and lab test results available to enhance our ability to predict sepsis, anticipate its onset time, and predict patients' future vital signs through a multitasking, multilayer ensemble neural network model. The proposed ensemble model is expected to have better performance than all its individual members because it is no longer necessary to tune each individual neural network to ensure the optimal weights are found, in contrast to when using a single ANN model. To achieve these goals, first, we extracted each patient's data collected in the first six hours from their admission to the ICU that is recorded in the MIMIC III dataset. Certain preprocessing steps are applied to the extracted data, including strategies for handling missing data, outlier removal, and data balancing. Various ensemble neural network models were then trained on the features extracted from this data for the first six hours of each patient's admission to predict sepsis occurrence in that patient. Using the proposed model has given us various insights that can be summarized as follows. (1) The definition of sepsis 3 is impractical in terms of real-world clinical practice. (2) Successful prediction of sepsis is associated with various vital signs and laboratory tests rather than medical scores. (3) Using the relevant vital signs as inputs improves the prediction accuracy regardless of the algorithm used. (4) Statistical features that are derived from a patient's time series data can be considered more useful than those from baseline data. (5) The classification results endorse the idea that ML models' discriminative power can be utilized to redefine how we classify and identify sepsis by relying on various clinical markers. This is a different approach whose definition of sepsis does not fully align with the classical definition of Sepsis that is currently in use. (6) Our results demonstrate that using the multitasking paradigm with a deep learning ensemble architecture can contribute effectively to improved model performance and act as a regularization step. The proposed model makes the following contributions:

- We propose a multilayer ensemble model that predicts sepsis in ICU patients while at the same time providing prediction of onset time and the blood pressure at that time.
- The first layer of our system is a classification model implemented as an ensemble of deep learning models; the second layer is an ensemble of deep learning regression models for multitask learning.
- To the best of our knowledge, there are no studies in the literature that are able to predict sepsis by giving its expected onset time while also offering medical proof for these predictions by providing an addition verifiable data point of the expected blood pressure at that time.
- We utilized NSGA-II, which is a well-known multi-objective feature selection optimization technique based on genetic algorithms to select the smallest number of features that can achieve the best performance.
- Medical experts guided our study in terms of directing us to select the initial set of relevant features that are medically trusted for the diagnosis of sepsis.
- Our model was implemented and tested using a large population from the MIMIC III dataset.
- The proposed model is statistically compared to other classical machine learning models from other studies in

the literature, our model achieved superior performance to all those models.

The remainder of the paper is organized as follows. Section 2 is the Related Work Section. Methods and Materials are detailed in Sect. 3. Section 4 details the proposed framework. Results and Discussions are given in Sect. 5. Study limitations are discussed in Sect. 6, and the paper is concluded in Sect. 7.

2 Related work

2.1 Medical scoring systems

In 1991, early efforts were developed to predict sepsis based on the SIRS criteria [29]. Four factors were specified as the SIRS criteria. Figure 2 details the SIRS criteria. To be diagnosed with sepsis 1, patients must meet two or more of the SIRS criteria. Hug et al. [30] stated that transient hypotension that is detected from blood pressure waveforms could later lead to sepsis, which, in turn, leads to an increased mortality rate. Wei et al. [31] stated that changes in heart rate and blood pressure are associated with the onset of decompensation and deterioration in critically ill patients. Despite this, various studies have acknowledged that the SIRS criteria are not accepted for defining sepsis [32]. Kaukonen et al. [33] analyzed the SIRS criteria for 109,663 patients with organ dysfunction, they found that 22% were classified as SIRS negative (i.e., SIRS criteria < 2). Moreover, the SIRS criteria were often present in patients who did not develop any infection [34, 35].

In 2001, a task force confirmed these drawbacks to sepsis 1, but they did not expand the list of diagnostic criteria or provide alternatives [9]. Therefore, sepsis 2 was introduced to define which patients had sepsis using the same criteria as sepsis 1. In 2016, as a part of the society of critical care medicine (SCCM) and the European society of intensive care medicine (ESICM) [36], a task force

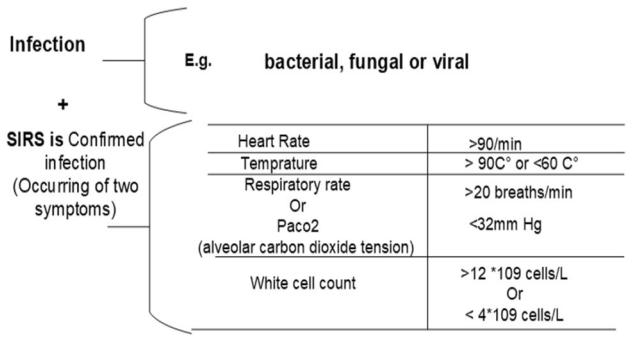

Fig. 2 The Systemic Inflammatory Response Syndrome (SIRS) criteria. Patients are classified as having sepsis if two or more SIRS criteria exist with a suspicion of infection

compared the SIRS criteria with other assessment scores (i.e., sequential organ failure assessment [SOFA]). Using SOFA scores for sepsis prediction was found to be more valid and superior to the SIRS criteria in terms of AUC (0.74 vs. 0.64). Appendix 2, Table 12 details the calculation of SOFA scores. However, the complexity of calculating SOFA scores, along with the lack of the patient data required to do this, has led to late identification of sepsis when relying on SOFA. At the end of 2016 [37], quick SOFA (qSOFA) was introduced to alleviate this limitation of the SOFA score approach. qSOFA is considered an enhanced version of SOFA; see Appendix 2 Table 13. [38]

2.2 Machine learning techniques for sepsis prediction

Machine learning techniques have previously been used to predict sepsis onset [39]. In previous decades, various studies have tried to understand the relationship between sepsis and patient data, including looking at the deterioration of various vital signs and at lab test results [40], 41]. Some studies have shown that use of continuous heart rate and blood pressure data can provide promising results in terms of sepsis prediction. Shashikumar et al. [29] utilized frequently recorded time Ω-series data, including blood pressure (BP) and heart rate (HR). In [13], the authors used the Insight ML model to predict sepsis based on a set of clinical variables such as age, gender, vital signs, etc. That study used data from 22,853 ICU stays and achieved an AUC of 0.781 using fourfold cross-validation. For predicting pre-shock state, Liu et al. [42] and Kam et al. [20] both used collections of features (e.g., arterial pressure, heart rate, labs, risk scores including Glasgow Coma Scores (GCS) and Sequential Organ Failure Assessment (SOFA) scores, as well as respiratory rate) along with lab test results leading to reported AUC scores of 0.93 and 0.929 using tenfold cross-validation, respectively. Kam et al. [20] built an LSTM model, and Liu et al. [42] built an RNN model for this task. Liu et al. [42] discovered that serum lactate was the primary predictor for septic shock. Some studies used hemodynamic measurements derived from recorded waveform or discrete electronic health data. Ghosh et al. [43] used three waveforms: mean arterial pressure, heart rate, and respiratory rate to derive hemodynamic predictor variables. That study used 1,310 samples. Liu et al. [42] and Kam et al. [20] used discrete measurements. The model by Scherpf et al. [44] predicted sepsis 3 h prior to its onset with an AUC of 0.81. To do this they built an RNN model to predict sepsis using demographics, vital signs, and lab test result features, they reported an AUC of 0.81 using fourfold stratified cross-validation. Fagerström et al. [45] built an LSTM model using demographics, vital signs, lab results, and GCS

features to predict sepsis. The study was based on data from 59,000 ICU patients. The model achieved an AUC of 0.8306 using sixfold cross-validation. Song et al. [46] predicted sepsis within a 48-h windows based on demographics, vital sign data, blood gas estimations, blood cell counts, and pH levels using logistic regression algorithm. The study was based on 7870 patients, and the authors achieved an AUC of 0.861 using tenfold cross validation. Yao et al. [47] predicted sepsis using XGBoost and achieved an AUC of 0.835 with fourfold cross validation. That study was based on 3713 patients.

In [48], the authors validated the use of PCR analysis and neural network genes to predict sepsis in 92 ICU patients. That study achieved an AUC of 83.09%. In [49], Lukaszewski et al. studied the use of lab test results and biomedical signals to predict sepsis onset using a support vector machine (SVM). Their model could predict sepsis 0–24 h prior to the onset time, the study was applied to 1,239 ICU patients. The results of this study are not reliable because the data used were highly imbalanced. It only contained data on 16 patients with sepsis (i.e., 2% of the total dataset). The study achieved an AUC ranging from 0.30 to 095. In [50], the authors proposed two models, one for detection and the other for prediction of sepsis. The models predicted sepsis four hours before the onset of the disease. They explored multilayer perceptron (MLP), XGBoost, random forest (RF), and logistic regression approaches. The study reported that RF achieved the best performance with an AUC = 0.97 for sepsis detection and an AUC = 0.90 for prediction. In [11], the authors proposed a gradient tree boosting model for predicting sepsis 3 h prior to its onset based on an algorithm called *Insight*. This algorithm was based on nine vital signs extracted from patient's data recorded during admission. This model had been trained on 1,394 patients where 91.6% of patients had sepsis and 8.4% did not. The study reported an AUC of 83.0%. In [51, 52], the authors used the *Insight* algorithm to detect severe sepsis and got an AUC of 89.0%. In [12], the authors checked the validity of *the Insight* machine learning algorithm when predicting both sepsis and septic shock using an aggregated data set from the University of California. The authors carried out training and testing using the MIMIC III dataset. They then applied transfer learning to their classification model. The study concluded that the *Insight* algorithm outperformed other scoring systems such as SOFA, QSOFA, and MEWS. The same idea has been applied in [52]. Note that most previous studies have used either the MIMIC II or MIMIC III datasets [53].

Other studies have utilized various deep learning methods for sepsis prediction [54–56]. For example, Chen et al. [57] developed an extensible model for sepsis, they used 142 features extracted from patients' vital signs, demographic data, and laboratory tests. They reported a utility score of 0.472 and an AUC of 0.83. In [20], the authors used a long short-term model (LSTM) to make predictions sepsis would occur in the future in certain patients, this model achieved 92.2% in terms of AUC. Other studies have tried to solve problems related to complex decision boundaries using ensemble classifiers that learn from the nonlinear boundary. For example, in [58], the authors developed a model based on an ensemble of five LSTM models, each of which was trained on a different dataset before being combined to get a single probability for each patient. Others built a model based on an ensemble of five XGboost classifiers [59]. He et al. [8] proposed an ensemble model based on a set of deep and artificial features from the MIMIC III dataset. That study used forty clinical variables (i.e., eight vital signs, 26 laboratory values, and six demographics). For every individual, these features were measured and recorded once an hour for six hours. These features were processed by three simple LSTM models to extract deep features. These deep features were combined with the original raw features plus features related to SOFA and SIRS scores. The resulting collection of features were used to train an XGBoost and gradient boosting decision tree model to perform a binary classification task (giving a result of either 0 or 1) for sepsis mortality. The study achieved a sensitivity and specificity of 0.641 ± 0.022 and 0.844 ± 0.007, respectively.

In [60], Chang et al. proposed an LSTM model to predict the onset time of sepsis. The main idea behind this model was to use time encoding to solve the problem of data being recorded at irregular time intervals, they achieved an AUC of 0.892. Similar idea has been implemented in [61, 62], but the main difference with these papers was that they worked in two modes. The first mode used demographic and vital signs, while the second mode used vital signs, demographics, and laboratory test results. Mode 1 and mode 2 achieved AUC scores of 0.89 and 0.92, respectively. LSTM and a CNN have been used in [63] to predict extreme sepsis and septic shock. The sample data used for this study included 40,336 cases from the MIMIC III dataset. That data included the onset time of sepsis, vital signs, demographics, and laboratory tests for each case. The classification models reported AUC scores of 0.89, 0.88, and 0.87 for predicting sepsis 4, 8, and 12 h before its onset.

Kim et al. [64], developed a DL model known as SERA that is based on clinical notes. NLP and feature selection techniques were used for analyzing those clinical notes. They reported values of 0.87, 0.87, and 0.94 for sensitivity, specificity, and AUC, respectively. In [65], the authors used SOFA scores to predict sepsis. They utilized a CNN and RF model to predict SOFA scores based on data from 5,154 patients. They reported an AUC of 0.842 in terms of sepsis classification MAE, an RMSE of 0.659, and 1.23 to

predict the onset time. Yuan et al. [66] utilized XGBoost to predict sepsis among adults. They collected data from 1,588 patients (444 with sepsis and 1444 without sepsis) and reported an AUC of 0.89. In [67], they used the XGBoost algorithm to predict sepsis among children.

In [68], Ngufor et al. used multitask learning to understand both mortality rate and length of stay (LoS) among patients in the intensive care unit (ICU). Other researchers in [68] used a convolutional neural network model to handle several natural language processing tasks (i.e., language modeling, named entity recognition). However, none of these models dealt with problems related to data with diverse sequential structures [69–71]. For more information about the role of ML in sepsis management, readers are guided to the following surveys [72–74]. Although previous sepsis studies have achieved qualified successes in the early prediction of sepsis, more effort is required to achieve even more accurate performance and earlier predictions. Moreover, further efforts are needed to evaluate the effectiveness of less commonly used ML algorithms in sepsis prediction. Ultimately, owing to the dynamic nature of physiological systems, more studies are required to analyze the longitudinal time series data required for accurate sepsis prediction. Other studies have concentrated on building models for sepsis prediction under special circumstances. For example, Xie et al. predicted sepsis among patients with kidney disease [75]. Other researchers in [67] built an ensemble model to predict sepsis among children.

3 Materials and methods

3.1 Data set

MIMIC III is an ICU database developed by MIT lab [53]. It comprises electronic health record (EHR) data for patients admitted to various ICU units in a large tertiary hospital in Boston. MIMIC III comprises of data from 53,423 patient admissions which were aggregated in the period between 2001 and 2012. The MIMIC III tables include 4579 different measurements and 380 laboratory test results per patient. Table 1 shows the distribution of patients according to care unit type. Privacy issues were tackled by removing all personal patient data like names, phone numbers, etc. Various modalities were included in MIMIC III, these were laboratory tests, physiological tests, diagnosis details, as well as nursing notes and reports. The data are distributed as a group of CSV tables mapped to a PostgreSQL relational database. All tables are linked using unique identifiers such as SUBJECT_ID, ADDMISSION_ID, etc.

Table 1 Distribution of patients according to care unit type

First care unit	Admission type	Survived	Deceased
SICU	Emergency	2117	202
SICU	Urgent	65	15
MICU	Emergency	6387	1657
MICU	Urgent	104	34
CSRU	Emergency	2115	624
CSRU	Urgent	152	13

3.2 Sepsis definition

In this paper, we follow the sepsis definition declared in [11], which is known as the gold standard. The sepsis gold standard is defined as suspected infection paired with two or more SIRS criteria. To identify a patient as positive for sepsis, we depend on ICD-9 code 995.51, which is stored in the MIMIC III dataset. MIMIC III does not include the time of the sepsis diagnosis. Therefore, we defined the onset time as the hour in which two or more SIRS criteria first occurred. The SIRS criteria are detailed in Fig. 2.

3.3 Comparators scores

To ensure the effectiveness of the proposed model, we compared the best performance of each model with two commonly used scoring systems: the sepsis-related organ failure assessment (SOFA) [76] and the modified early warning score (MEWS) [77]. To calculate SOFA, we used patient measurements (including Glasgow coma score, PaO2/FiO2, bur. bilirubin level, etc.), where each measurement ranges from 1 to 4, and the overall score is calculated as the sum of all scores. Table 12 in Appendix 2 details the calculation of the SOFA score. The MEWS ranges from 0 to 14 and is scored by evaluating the patient's measurements, including heart rate, Glasgow coma score, temperature, respiratory rate, etc. The details of these subscores are presented in [77]. Table 14 in Appendix 2 details the calculation process for MEWS.

3.4 Feature selection

Feature selection is an important preprocessing step in classification task, its aim is to eliminate irrelevant, redundant, and noisy features. There are many feature selection techniques in the literature because there are many situations where the available data has hundreds of features leading to data with very high dimensions. To select the best features in these situations, a feature selection method can be efficient for removing irrelevant and redundant data, which should result in lower computation

time, improved classifier performance, and a better understanding of the final learning model and data. There are three main categories of feature selection techniques: using a filter, a wrapper, or embedding [78]. Each category has its own advantages and disadvantages. (1) Filter methods are based on ranking features individually before classification, a threshold is used and any features that are below this threshold are considered irrelevant and removed. The ranking methods are based on certain evaluation criteria, including correlation, dispersion ratio, variance threshold, relief, fisher's score, and mutual information. These univariate feature selection methods are fast, scalable, and independent of the classifier, but they neglect any correlation between features or interactions with the classifier. As such, these approaches can neglect relevant features that are meaningless by themselves but in combination these features may be able to improve model performance. (2) Wrapper methods are based on classifiers that act as black boxes where the classifier's performance informs the objective function to evaluate a subset of the feature set. This approach allows us to consider correlation among features so is not held back by dome of the limitations of filter methods [79]. However, wrapper methods are complex and more prone to overfitting when used with small training datasets [80]. As evaluating 2 N subsets is an NP-hard problem, optimal subsets can be found by using search techniques to find that subset heuristically. For example, the Branch and Bound method [81] uses a tree structure to evaluate different feature subsets. Exhaustive search methods are computationally intensive for datasets with many features. Therefore, some bio-inspired optimization techniques [82] such as genetic algorithms (GA), ant colony optimization (ACO), or particle swarm optimization (PSO) [83, 84] have been used to find local optimum results, which are sufficient to produce good results in a computationally feasible manner. Wrapper methods are classified as sequential selection algorithms (e.g., recursive feature elimination) or heuristic search algorithms (e.g., GA, PSO, etc.). (3) The third approach, hybrid embedded methods, incorporate feature selection as part of the training process. Both wrapper and embedded methods are classifier dependent. In [80], the authors studied the benefits and limitations of these three feature selection approaches. All previous methods are known collectively as supervised methods. There are many other feature selection techniques based on unsupervised, semi-supervised, and ensemble techniques [85]. The details of these techniques are provided in a published survey [80].

NSGA-II is a popular multi-objective optimization technique that has been used in many studies for feature selection [86–89], and in hyperparameter optimization tasks [90] because it has been shown to achieve better results than other state-of-the-art methods. NSGA-II can optimize an objective function with two conflicting objectives [91]. NSGA-II has shown great promise as one of the most efficient multi-objective evolutionary optimization approaches in the literature [92]. In detail, it has low time complexity of $O(NlogN)$ where N is the population size and can avoid difficulties while setting sharing parameters. NSGA-II also outperforms other algorithms for multi-objective optimization due to its lower computational complexity and its elitism property [93]. Salmanpour et al. [94] compared NSGA-II with other well-known optimization techniques (e.g., PSO, ACO, Simulated Annealing, etc.), they reported the best results came from NSGA-II which selected the smallest number of features while still achieving the best performance. Türkşen et al. [95] compared NSGA-II with other multi-objective optimization techniques for feature selection, including archived multi-objective simulated annealing (AMOSA) and direct multi-search (DMS) methods, it was again found that NSGA-II achieved the best results. Hojjati et al. [96] compared NSGA-II and PSO while they optimized the operation of two-reservoir systems with the goal of maximizing income from hydropower sales while providing effective flood control, they found that NSGA-II outperformed PSO. Zhang et al. [88] used the idea of the Pareto domination relationship and applied it using PSO, they managed to achieve comparable performance to NSGA-II in this task. Based on the previous discussion, we chose to apply NSGA-II as our feature selection technique.

3.4.1 Multi-objective optimization for feature selection

The data used while building the classification model includes a wide range of features that affect both classification accuracy and learning time. Therefore, it is important to select important features before building the classification model. Feature subset selection is the process of selecting a subset of features from the whole feature set according to specific optimization criteria [24]. The multi-objective genetic algorithm (MOGA) is considered one of the most sophisticated engineering optimization techniques for this purpose [97]. It includes various techniques such as the micro genetic algorithm (Micro-GA), strength Pareto evolutionary algorithm (SPEA), non-dominated sorting genetic algorithm II (NSGA-II), etc. NSGA-II [22] is a well-known optimization technique that has three main characteristics: (1) using the elitist principle which assigns the various probabilities of creating the next generation, (2) using the crowding distance and fast crowded comparison methods, and (3) emphasizes non-dominated sorting solutions [23]. In this paper, we use NSGA-II for feature subset selection due to its low time complexity [98]. NSGA-II has been used in many studies in different fields and has often achieved the best results [99, 100].

The algorithm uses non-dominant sorting and crowding distance to select the fronts of the population. Then crossover and polynomial operators are used to combine parents and offspring to generate the next generation. The best solution is selected based on diversity and non-dominant sorting. Algorithm 1 outlines the NSGA-II main steps.

3.4.1.1 Dominant ranking For an objective function, let M and N be two solutions. M could dominate N if it meets the following criteria: (1) M is not worse than N for all values of the objective function, and (2) M is superior to N in at least one value of the objective function. Otherwise, M and N are said not to dominate each other. Algorithm 2 details the steps for the dominance ranking.

3.4.1.2 Crowding distance The crowding distance is used to calculate the density of each solution. Consider Z to be a non-dominated solution with size S, and objective function $F_o whereo = 1, 2, 3, \ldots, \lambda$ is the crowding distance. The calculation of crowding distance is detailed in algorithm 3. To compare between two solutions *M* and *N* both dominant rank and crowding distance should be calculated.

Algorithm 3: Crowding distance.

1. Let $C_j = 0$ for $i = 1,2,3, \ldots, Z.$
2. For each objective function F_o $where$ $o = 1,2,3, .. \lambda$
3. Let C_j and C_z be the maximum values of the crowding distance
4. For $J = 2$to $(Z - 1)$, set $Cj = Cj + (F_{o+1} - F_{O-1})$

3.5 Ensemble of artificial neural network

An artificial neural network (ANN) is a type of artificial intelligence that attempts to approximate a given function by tackling it in the manner of a biological nervous system. An ANN is made up of several interconnected nodes called neurons. A traditional feedfoward neural network at minimum contains an input layer, one hidden layer, and an output layer. Each neuron in the hidden layer receives input data adjusted by weights from the previous layer, in addition there is a bias from each neuron, as follows

$$z_i = \left(\sum_{k=1}^{N_j - 1} x_k^{j-1} w_{k,i} - b_k \right) \quad (1)$$

Algorithm 1: The NSGA-II Algorithm.

1. [Start] Generate a random population of chromosomes according to problem range
2. [Fitness Function] Calculate the multi-objective fitness function for each chromosome *N*
3. [Ranking] The population members are ranked according to the following:
   - 3.1 [Dominant ranking] Population members sorted according to algorithm 2.
   - 3.2 [Crowding distance] Crowding distance calculated using algorithm 3.
4. [Next population] The new generation is created by repeating the following steps:
   - 4.1 [Selection] Select two chromosomes (parents) from the population according to the crowding distance algorithm (algorithm 3).
   - 4.2 [Cross Over] Cross over the parents to generate the offspring (children).
   - 4.3 [Mutation] Mutate the offspring.
   - 4.4 [accepting] Replace the new offspring with its parents.
5. [Replace] Use the new generation to rerun the algorithm.
6. [Test] Check if the stop condition is satisfied (i.e., the max number of generations is reached).
7. [Loop] Repeat from step 2.

Algorithm 2: Dominance ranking.

1. Start with rank r is equal to 0.
2. Increase r value$r = r + 1$.
3. Calculate the nondominated individuals for each population member (individuals)
4. Assign rank to all individuals
5. Remove ranked individuals from population p.
6. Stop when population $p$is empty, otherwise, go to step 2.

where x_k^{j-1} represents the input of the *k-th* node located in the *j-th* layer, and $w_{k,i}$ represents the weight between the k node in one layer and other nodes in the previous layers, while b_i represent the bias, $N_j - 1$ is the number of nodes in layer $j - 1$. Figure 3 shows the general architecture of a feedforward ANN. To produce its output, the summation is passed along to the activation function that is calculated as $Y_i = (Z_i)$. The most common activation function is the sigmoid function that is calculated as in Eq. 2

$$F(Z_i) = \frac{1}{1 + e^{-z_i}} \quad (2)$$

Although training deep neural networks may take time and resources, there is no guarantee that the final model will have a small generalization error [101]. On the other hand, neural networks are very sensitive to initial conditions (i.e., the initial weights and the existence of noise in the training dataset), problems here may result in a low bias, high variance model. One possible solution to these problems is the combination of multiple models. This approach belongs to a general class of techniques called ensemble learning [26, 102]. Figure 4 shows the concept of an ensemble of neural network models.

Multiple neural networks with the same configuration but different initial weights are trained on the same feature space. Each model makes its own prediction, and the final decision is calculated by taking the average of the outputs of all models. This solution helps to produce a low variance model, but it may not contribute to reducing generalization errors. This is because all models may have highly correlated errors as they are trained using the same mapping functions. Alternatively, the configuration of each neural network may have a different architecture (i.e., different learning rates, regularizations, numbers of nodes, numbers of layers, etc.). The final output is calculated by combining the predictions of all neural network models using various combination techniques. They may be combined by weighting the prediction of each model, this is known as blending. Another approach is using a new model that learns how to best combine the output from each ensemble member, this is known as stacking.

3.6 Multitask learning

The purpose of MTL is to integrate the learning of several related tasks to enhance the training process and improve model performance. In [103], the authors provide a detailed description of MTL. In [104], the authors provide a comprehensive review of MTL in relation to deep learning. MTL helps a model to differentiate between relevant and irrelevant features and draws its attention only to the features that affect the tasks at hand. MTL makes the model produce a more suitable representation for all tasks, which helps the model to generalize all its tasks. Finally, MTL reduces the risk of overfitting by acting as a regularizer through inductive bias.

In the clinical space, MTLs have been used in various frameworks of prediction and regression models. In [105], Chen et al. proposed a multitask CNN and RNN to predict mortality. In [106], Harutyunyan et al. used MTL to predict ICU mortality, LoS, phenotyping, etc. In [107], Wang et al. compared the performance of single task and multitask models to demonstrate the effectiveness of transferring knowledge among related tasks. El-Sappagh et al. [16] introduced a multitask deep learning model that was based on BiLSTM and a CNN for multiclass classification of Alzheimer's disease.

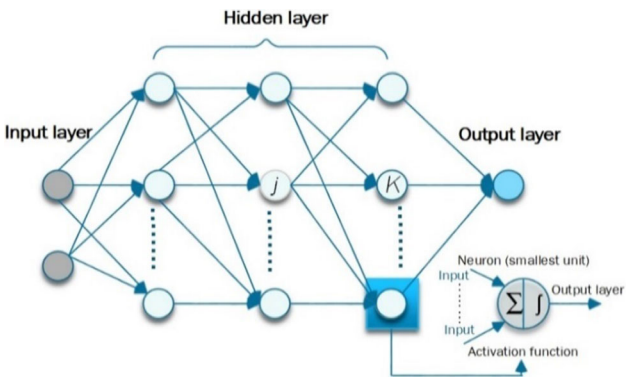

Fig. 3 Simple neural network

4 Proposed framework

In this study, we develop a prediction model for patients at risk from sepsis. This model is divided into two layers. Figure 5 shows the general architecture of our framework. The first layer is a classification model used to predict patients who may develop sepsis based on the data extracted from their data collected in the first 6 h after their ICU admission. Single and ensemble deep learning models were optimized and compared in terms of their detection performance in this layer's task. We explored different models and architectures with various features sets using different optimization techniques. The second layer is the regression model, it is used to predict the onset time at which a patient starts to develop sepsis and predicts their blood pressure at that time. We chose to predict blood pressure due to its importance for sepsis patients during the treatment process. In the second layer, we initially utilized different machine learning regression models for prediction (including linear regression (LIR), lasso regression (lasso), ridge regression (ridge), SGD regression (SGD), random forest regressor (RF), gradient boosting regressor (GB), and decision tree regressor (DT)). Then, we developed both single task and multitask deep learning models for prediction. Through this we demonstrated that using MTL for simultaneously predictions of related task increases the knowledge transfer between tasks and improves the overall learning process. All models in the second layer were evaluated using both MAE and RMSE.

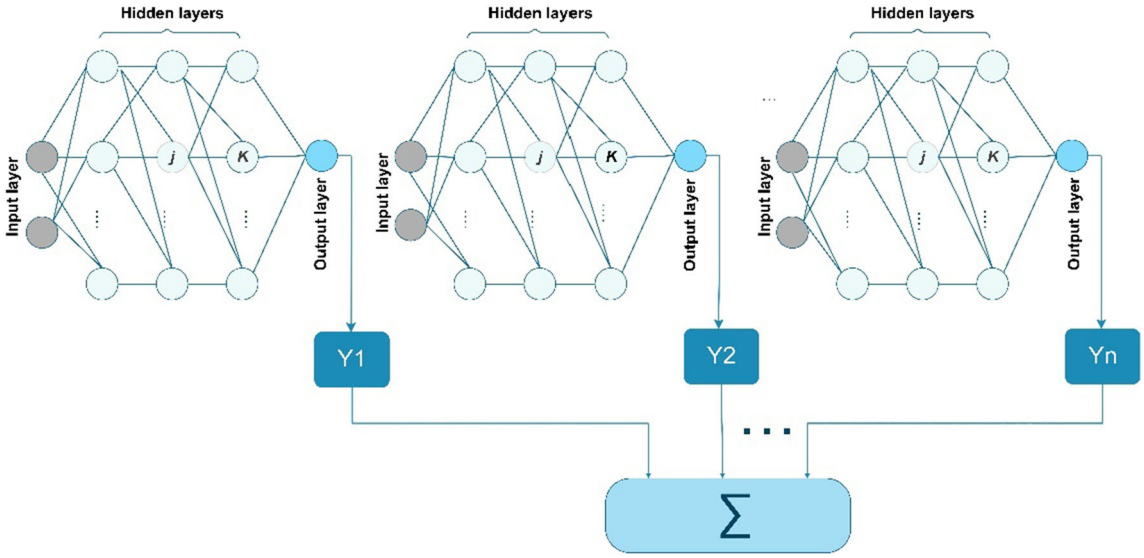

Fig. 4 Ensemble neural network

4.1 Data inclusion criteria

Data in MIMIC III is temporal, and most entries are part of a time series. Some fields are updated over time periods of hours and others are updated over minutes. Patients were monitored from the time they were admitted to an ICU unit ($t = 0$) until the time the patient was discharged. In MIMIC-III, 38,597 distinct patients were aged over 15, 16,1273 patients were diagnosed with sepsis (ICD-9 code = 99,591), 3913 patients had severe sepsis (ICD-9 code = 99,592), and 2857 patients suffered septic shock (ICD-9 code = 78,552). In our study, we concentrated on predicting sepsis in all patients (male and female) who were older than 15. Figure 6 details the selection criteria for this study.

- We included adult patients (age $>$ 15) admitted to any medical ICU unit.
- We included patients who were not diagnosed with SIRS at the time of admission, or within the first 24 h after that admission.
- We included patients that had at least 1 value in each measurement category for sepsis patients as well as patients that had at least 2 values in each measurement category for healthy patients.
- We excluded a random set of records from the majority class (i.e., healthy class) to make the data balanced.

To tackle the challenge of the sepsis onset time not being mentioned in the data set, a label was created for each hour of patient admission. This label indicates either 1 (Positive for sepsis onset) or 0 (Negative for sepsis onset). Positive and negative labels are defined according to the patient's vital signs in each hour. A patient is considered positive for sepsis if two or more SIRS criteria occur simultaneously. The SIRS criteria are discussed in detail in Sect. 3.2.

4.2 Data preprocessing

This step aims to improve the quality of the extracted data. After exploring the extracted dataset, we found that it contained many outliers and missing values. Missing values may occur for various reasons including sensor failure, network transformation error, etc. Training a model using incomplete and noisy data is recognized as one of the main routes to poor performance in machine learning [108]. The data preprocessing step includes tasks such as handling irregular time intervals, data balancing, removing outliers, and handling missing values.

4.2.1 Handling irregular time intervals

Most vital signs are measured at irregular time intervals. Often machine learning techniques are not designed to work with time-series data. Although some of them can be adapted to use streaming data, they often require the data to be sampled at regular time intervals. To solve this problem, we divided each patient's 24 h stay into 24 sequential intervals each with a length of one hour where one value for each data point is assigned to each interval, this is achieved by averaging measured data over that hour period for quantities that are sampled often. As a result, each record contains 24 different values for each datapoint to cover the 24-h period of the study. For the hours without observations, the missing values are taken from the nearest

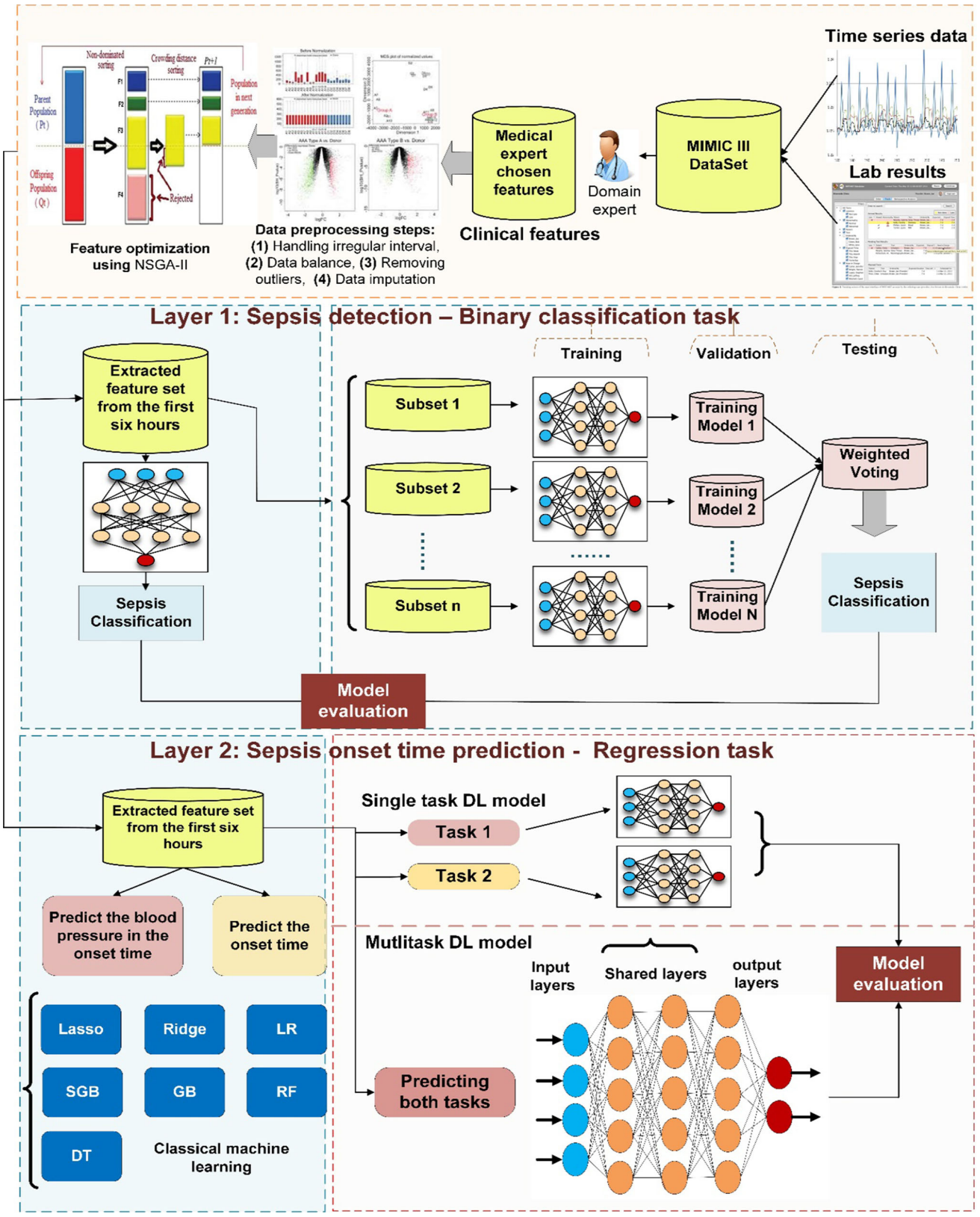

Fig. 5 The proposed framework

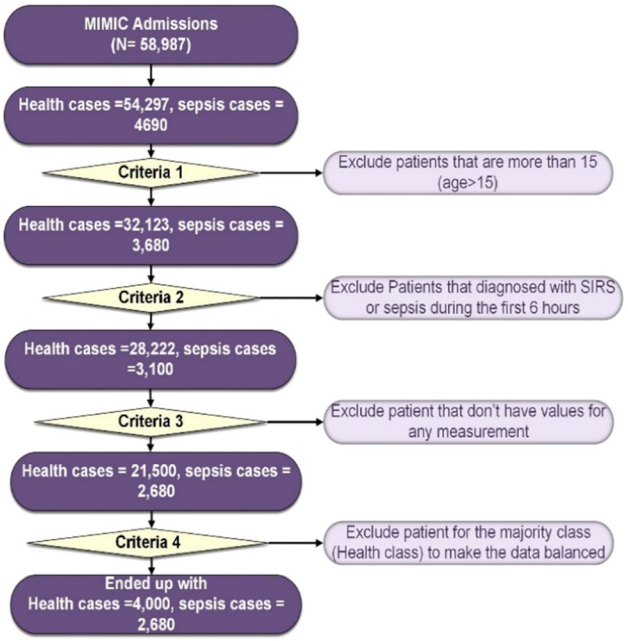

Fig. 6 Data inclusion and extraction criteria of the patients used in the study

available observations. All these calculations were carried out using python scripts.

4.2.2 Data balancing

A common issue with medical data is class imbalance [109]. MIMIC III is an imbalanced data set where the minority class is the patients with sepsis. Most ML techniques do not work well with highly imbalanced datasets as it causes the ML algorithm to become biased for one class and classifies all data into the majority class. The three popular techniques for handling imbalanced data are downsampling, oversampling, and a combination of oversampling and downsampling [110]. Oversampling increases the number of records in the minority class. Downsampling works by decreasing the number of records in the majority class. Various algorithms can be used for oversampling such as random oversampling [111], the Synthetic Minority Oversampling Technique (SMOTE) [112], the adaptive synthetic sampling approach, or ADASYN [113]. Common downsampling techniques are random under-sampling [113], clustering and the Tomek links methods [114]. In this study, we use various data inclusion criteria including age, the existence of at least 3 values for each measurement, and data balancing. After excluding data according to the first two criteria, 35,445 patients were included. Using the downsampling technique does not add any noise to the data but excludes some records from the majority class. In the final analysis, we noticed that downsampling contributed to improved results.

4.2.3 Removing outliers

Outliers are defined as values that lie too far from the normal range. The opinions of medical experts are used to characterize the normal range for each measurement. The values that lie at an abnormal distance from the normal range were eliminated and imputed using the expectation–maximization algorithm [115].

4.2.4 Data imputation

The existence of missing values is a common problem in medical data. This is because the values in the dataset may have been recorded or sampled at varying time intervals [116]. A simple way to handle missing values is to exclude records that contain missing values. However, this results in removing a significant portion of the data. Therefore, various data mining practitioners and researchers have done extensive work to address this problem by exploring different approaches to handling missing values such as expectation maximization [115], hot-deck imputation [108], and multiple imputations by chained equation [117]. In the analysis of multivariate time series data from MIMIC III, a large proportion of the laboratory test and vital sign data are missing at different times during patient admission. For example, time-series data such as temperature and blood pressure (invasive and non-invasive) comprise of between 45 and 60% of the lost data, but we could not eliminate these datapoints due to their importance in the detection process. Considering this disparity, we first choose sepsis cases that had at least 2 values for each measurement and choose normal cases that had at least 1 value for each measurement. The expectation–maximization algorithm was used to impute missing values in both sepsis and normal cases [115].

4.3 Feature selection

In this study, our feature selection process was conducted in four main stages. *First*, we reviewed and analyzed the features used in previous literature. *Second*, we consulted a medical expert to recommend the most critical features for sepsis prediction from a medical point of view. The first two stages resulted in extracting the 36 most important features (vital signs and laboratory measurements) that may be used as inputs to our model. *Third*, we calculated a statistical feature for each measurement per hour. *Fourth*, these statistical features were optimized using certain feature optimization techniques we selected.

4.3.1 Feature extraction

In this step, we depended on both previous studies and medical expert opinion when choosing the most important features that help predict sepsis among patients with the most common diseases and identified the effect of sepsis in each measurement [10, 29]. For example, for a diabetes patient, sepsis increases the blood glucose level. On the other hand, for non-diabetic patients, sepsis decreases the glucose level to be less than average, this may cause the glucose levels to reach those of hypoglycemia [118]. For patients with hepatic diseases, sepsis increases SGOT, Alkaline Phosphatase, burlibun enzyme, and Cerataine while it decreases Albumin. For patients with renal diseases, sepsis raises the renal failure probability which is characterized by decreases in Pao4 and lactate levels. In addition to this, sepsis causes increases in arterial base excess which induces metabolic acidosis (blood PH $<$ 7.35, HCO3 $<$ 20) [119]. For patients with low blood disease, sepsis can progress in a way that may lead to disseminated intravascular coagulation (DIC) in addition to dysfunction in platelet count and function according to white blood cells (WBC). Urine is also critical in predicting sepsis, patients with a risk of sepsis suffer from decreased urine output $<$ 0.5 ml/kg per hour [118]. The chosen features and their normal ranges are clarified in Appendix 1, Table 11. We demonstrate the utilization of several measurements for sepsis prediction. A total of 36 features were chosen for our model, this will not only increase performance in prediction but also offer better interpretability for clinicians at the level of the input variables which may help in specifying the cause of any dysfunction and help in the development of a therapy plan [11, 44, 120]. For each patient admission, we only extract data from the first six hours in ICU. The reason for choosing only the first six hours is related to developing a prediction model that can predict sepsis as early as possible and will help avoid sepsis progression. Figures 7, 8 shows the feature extracting process according to the time frame that we used to handle the challenge related to having various features (heart rate, respiratory rate, etc.) that have several measurements recorded during the same hour. Statistical functions (including minimum, maximum, average, standard deviation, and variance) were calculated for each feature in each hour. This step ended giving a total of 1080 features (36 feature*5 statistical functions*6 h = 1080 features).

4.3.2 Feature optimization

In this step, feature optimization is conducted to choose the optimal feature subset from the whole feature set (that includes 1080 features). Two competing objective functions were used to choose the optimal feature subset. First, we utilized NSGA-II to choose the minimal number of features from the extracted dataset. The list of NSGA-II parameters can be found in "Appendix 3" Table 15.

The principle of NSGA-II is to use non-dominant sorting and the crowding distance to choose different feature subsets. NSGA-II was discussed in detail in Sect. 4.1. Second, the classification error, we used the 1-Neural Network (1-NN) as the classification model to evaluate the performance of each feature subset identified in the first step. The 1-NN was built based on training data and denotes classification errors based on testing data with a fitness value for each feature subset specified using the NN. Then, the feature subset that gave the lowest error rate was chosen as the optimal feature subset. Figure 6 details the steps for choosing the best feature subset. Next, we applied NSGA-II to this feature set to extract the optimal feature. We end up with 660 features after applying this feature selection technique.

5 Results

5.1 Experimental setup

All experiments in this paper were implemented with an intel core i7 laptop workstation with 16 GB Ram and a 1 Terabyte hard disk under a Windows 10 64-bit system. We used Python 3.7 distributed with Anaconda 5.0.0. All

Fig. 7 Time frame extraction, only the first six hours are used as an input to both the classification and the regression layer, the rest of the data from the patient's stay was used to confirm the prediction time. Note that we leave 2 h as a time gap between the training and testing window to maintain a reliable and confidential model

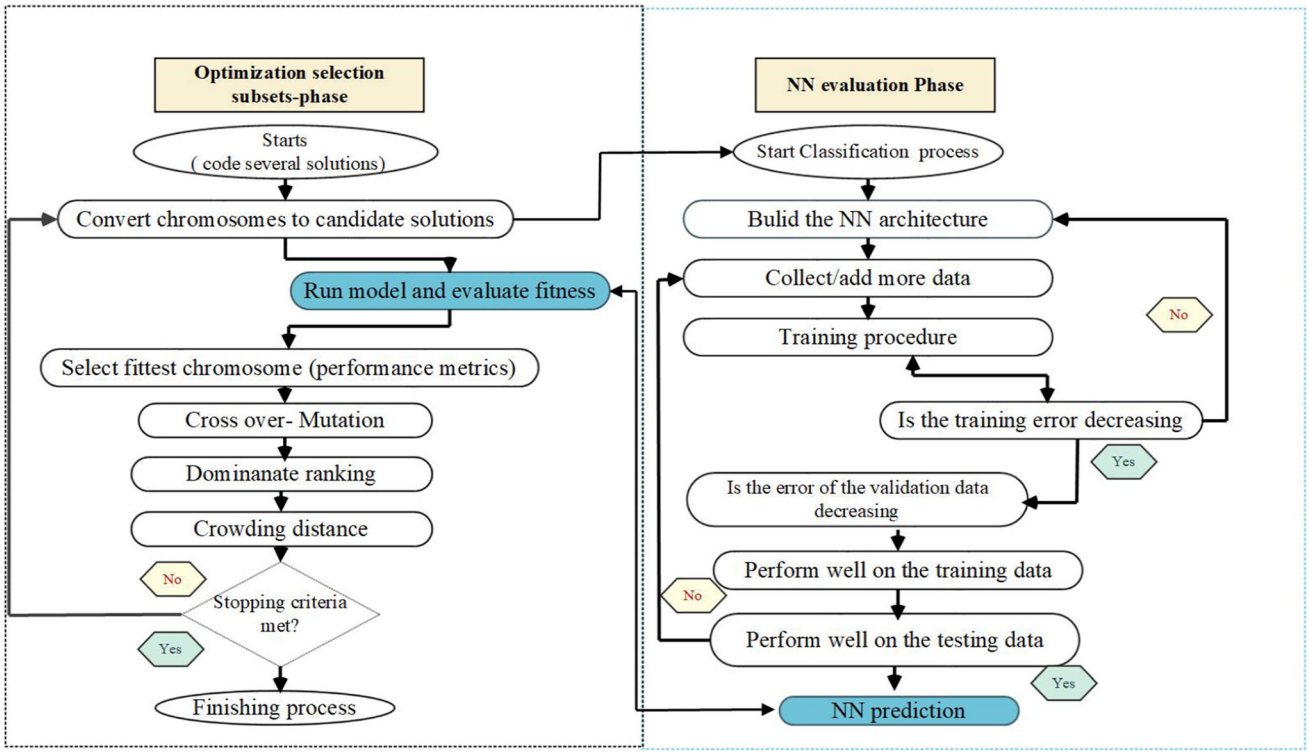

Fig. 8 Hybrid NSGA-II and neural network

models were implemented using the Keras library that is based on the TensorFlow backend. The SoftMax activation function with binary cross entropy was used in the classification layer, while the linear activation function was used for regression tasks.

To evaluate the effectiveness of the proposed model, we implemented and tested various DL models using a simple neural network, an ensemble stacked neural network with two different meta-classifiers (NN and logistic regression (LR)), and an integration of single and ensemble models with NSGA-II for feature subset selection. We found that the stacked ensemble model outperformed all other models, and accordingly used this in our classification tasks. Then, we utilized single task and multitask DL models to predict the values for both sepsis onset time and the patient's blood pressure at that time, our results demonstrate the ability of the multitask model to enhance overall performance when compared with single task DL models.

5.2 Evaluation metrics

For classification task, we used three metrics include classification accuracy, sensitivity, specificity, and AUC. The cross-validation (CV) results are calculated based on the training data, and the generalization performance is measured based on the testing data. Tables 2, 3 details the used evaluation metrics.

5.3 Results for the classification layer

A goal of this layer is to identify who may obtained sepsis at any time in their first day in the ICU admission after the first six hours. Several experiments are conducted using single and stacked ensemble of DL models. These models are explored with and without feature optimization step. All classification models are tuned using the Bayesian optimization [121] and grid search [122] techniques.

5.3.1 Model training

We propose an advanced DL model for detecting Sepsis, and it utilizes the patient's time-series data to predict sepsis based on multiple features. First, for the classification task, we fed the patient's features (Sepsis, no sepsis) into a pipeline of DNN and dense block. This block has the following layers: (1) input layer with a dimension of 263, (2) A rectified linear unit (ReLU) activation function (3) Four separate dense layers with a different number of neurons, (4) L2 regularization equal to 0.01 (5) dropout layer with percentage 0.1 (6) the final layer for the classification problem uses the SoftMax activation function with binary cross-entropy. All classification models were trained using Adam for multi-objective loss function with learning with 100 epoch and batch size of 30. Furthermore, to avoid overfitting, we used L2 regularization with parameter 0.01.

Table 2 Evaluation metrics

Metric	Abbreviation	Equation	#	Definition
Accuracy	*ACC*	$\frac{tp+tn}{tp+fp+tn+fn}$	(3)	The percentage between number of cases that are correctly classified and the total number of cases
Specificity	*SP*	$\frac{tn}{tn+fp}$	(4)	The percentage of the negative class cases that classified correctly
Area under the ROC curve	*AUC*	$\frac{s_p-n_p+(n_{n+1})/2}{n_p n_n}$	(5)	It measures the ability of the model to discriminate between classes. Where sp is the number of cases in the positive class, nn and np is the number of negative and positive class respectively
Mean Square error	*MSE*	$\frac{1}{N}\sum_{i=1}^{N}(y_i - y_i)^2$	(6)	Measure the differences between values (sample or population values) predicted by a model or an estimator and the values observed
Mean absolute error	*MAE*	$\frac{1}{N}\sum_{i=1}^{N}\lvert y_i - y_i\rvert$	(7)	Measure the closeness of the prediction to the eventual outcomes

Table 3 Results of single classifiers

Classifier	Feature optimization	Classifier optimization	CV accuracy	ACC	Sn	Sp	AUC
Single DNN	–	–	0.747 ± 0.02330	0.677	0.756	0.714	0.730
	–	GS	0.7562 ± 0.0119	0.756	0.759	0.726	0.740
	–	BO	0.758 ± 0.02370	0.750	0.769	0.725	0.752
	NSGA-II	–	0.791 ± 0.01190	0.793	0.824	0.793	0.787
	NSGA-II	GS	0.800 ± 0.01190	0.803	0.844	0.761	0.791
	NSGA-II	BO	0.816 ± 0.09800	0.802	0.783	0.753	0.801

We split our dataset into stratified 70% for training, 15% for validation and 15% for testing. To avoid bias, we used the stratified tenfold cross-validation technique. *Note that* all experiment repeated using the following factors (single and ensemble classifier, with and without choosing the optimal feature set (NSGA II). The classifier hyperparameters optimized using both grid search optimization (GSO), Bayesian optimization (BO) techniques, and tenfold cross-validation by varying batch size and learning rate.

5.3.2 Single classifier

In this section, we utilize a single DL model for predicting sepsis after the first six hours of the patient's admission, and it is considered a binary classification task. We conduct six different experiments to check the performance with various conditions (with and without feature optimization, with and without classifier optimization). Firstly, the model was build based on all patient's features (1080 features). It results the lowest performance (ACC = 0.677, AUC = 0.730). The performance was slightly improved when optimized the classifier hyperparameter (i.e., classifier learning rate (CLR), number of epochs, dropout percentage) sing BO and GSO (ACC = 0.756, AUC = 0.740). The optimized hyperparameters can be found in "Appendix 3". Using the optimized feature set (660 features) increase the performance by about (6–12) %. The best performance was obtained when the classifier was tuned using BO (ACC = 0.802, AUC = 0.801). To enhance the performance, in the sext section, we utilize the stacking ensemble deep learning model.

5.3.3 Proposed ensemble classifier

In the stacking ensemble algorithm, n subsets of the training set were created using the stratified with replacement technique, where relative proportion from each class is maintained in each subset. We test two meta-classifiers include LR and NN. The optimized hyperparameters can be found in "Appendix 3". The use of ensemble deep learning models in the prediction of sepsis was validated based on the data of the first six hours after the patient's admission. First, we start with building a classification model that utilizes the whole feature set (1080 feature) in building the ensemble DL model, and we obtain ACC = 0.727 and AUC = 0.781. As we expected, using the optimized feature set (660 feature) that chosen using NSGA-II enhanced the classification performance by about 5% (i.e., ACC = 0.865 and AUC = 0.861). To improve classifier performance, we used two optimization techniques include GSO and BO.

Using GSO enhance the performance by 0.01 and 0.02 in terms of ACC and AUC, respectively (i.e., ACC = 0.890 and AUC = 0.886). The best performance is obtained by using the Bayesian optimization technique for tuning the hyperparameters (CLR = 0.001, batch size = 128, dropout = 0.1) of the classification model (i.e., ACC = 0.913 and AUC = 0.906). Figure 9b details show the performance of all experiments with the ensemble model. The obtained results confirm the strength of our proposed system for predicting sepsis disease. To the best of our knowledge, our proposed system is the first algorithm that exceeds 0.90 for AUC based only on the data of the first six hours [52, 123, 124]. Additionally, it achieved superiority over traditional severity scores such as SOFA and MEWS that mainly used for screening sepsis. Our proposed model achieved superiority over state-of-the-art for various reasons include the following. (1) depend on an applicable definition that utilizes various measurements for various diseases (2) depending on several measurements that related to various types of diseases. (3) choosing the optimum feature set using the feature optimization technique. To conclude, the experiment results demonstrated that each feature has an important role in both the classification and the regression tasks.

5.3.4 Statistical analysis

To ensure a significant difference between all the simple and ensemble DNN models, all models were compared using the Friedman test [125]. The Friedman test is a non-parametric test used to determine if there is a significant difference between models without specifying which is best. To choose the best performance model according to statistical testing, the average rank for each model was calculated based on the Nemenyi test [126]. Results of the Nemenyi test can be visualized using a critical difference diagram. Figure 10 shows a comparison between classification models based on the critical difference calculated from the results of the Nemenyi test for all models. The test shows a significant difference between all models (Statistics = 6.34, $P < 0.005$). Figure 10 shows that using a single DNN without feature optimization gives the worst performance (i.e., AUC = 0.730, $P < 0.005$), next worse was the model with the same feature set after applying tuning with the Bayesian optimization technique. Using feature optimization increases the performance of the model (i.e., AUC = 0.791, $P < 0.005$). When using ensemble neural networks, the worst performance was obtained when using the whole feature set without feature optimization (i.e., AUC = 0.781, $P < 0.005$). Choosing the optimal feature set increases the performance (i.e., AUC = 0.861, $P < 0.005$) while using LR as the meta classifier and a further boost is given when (i.e., AUC = 0.865, $P < 0.005$) using NN as the meta classifier. Using the grid search algorithm enhances the classifier performance (i.e., AUC = 0.886, $P < 0.005$). The best-performing model was obtained when using the optimal feature set with the Bayesian-based optimized ensemble model and a neural network as the meta classifier (i.e., AUC = 0.906, $P < 0.005$). Table 4 shows all evaluation metrics for the ensemble models with various settings.

5.3.5 Comparison with scoring systems

To ensure the superiority of our stacking ensemble DL model, we conducted several experiments to compare it, in terms of AUC score, with the two common scoring systems used to identify sepsis: SOFA and MEWS. As mentioned earlier, SOFA and MEWS are commonly used in predicting sepsis. First, we calculated these scores based on the appropriate features and calculation approaches. Appendix 2, Tables 12 and 13 detail the calculation techniques for SOFA and MEWS, respectively. Note that all scoring

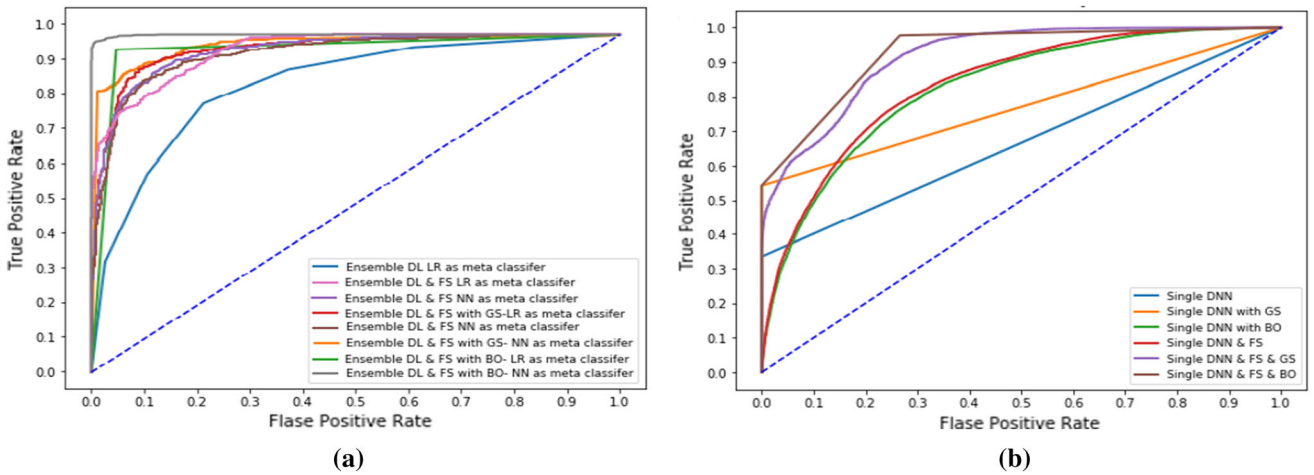

Fig. 9 (**a**) Results of single classification models. (**b**) Results of ensemble classification model

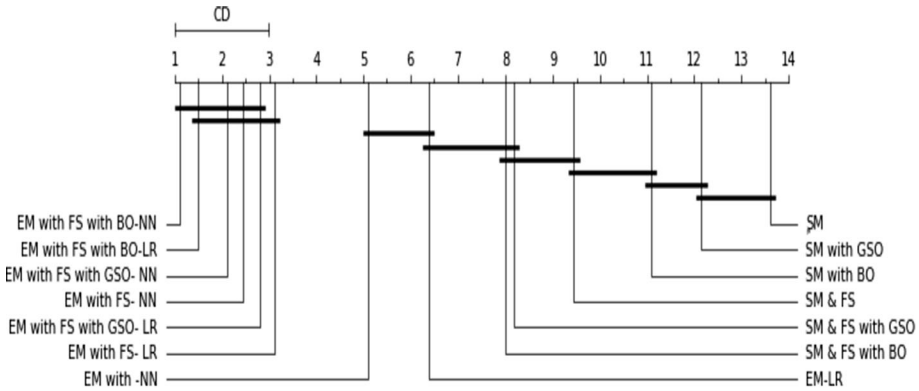

Fig. 10 SM: Single Model, GSO: Grid search optimization, BO: Bayesian optimization. FS: Feature selection (NSGA II), EM: Ensemble classifier, LR: refers to using Logistic regression as a meta classifier, NN: refers to using neural network as a meta classifier

systems calculated using data that had been preprocessed to provide a fair comparison. As shown in Table 5. SOFA achieves better results than MEWS (i.e., ACC = 0.740, AUC = 0.78). From the above experiments, we make the following observations. (1) Most scoring systems achieve proximate performance, which is not reliable enough to provide sepsis prediction, (2) the fusion between various features from several sources produces more accurate predictions compared to others models based on fewer features, (3) the deep stacked models is more robust and accurate than the traditional scoring systems, (4) using a stacked ensemble and optimized DL classification model enhances the performance compared to the single DL model, (5) statistical features that are derived from time-series data are more significant than baseline data. Figure 11 shows a comparison between the traditional scores and our model.

5.3.6 Comparison with standard ensemble classifiers

To explore the performance of other machine learning approaches and compare it with our work, in this step, we utilized state-of-the-art machine learning algorithms (i.e., standard ensemble classifiers). The ensemble classifiers tested included homogeneous classifiers (i.e., random forest (RF), bagging, and extreme gradient boosting (XGBoost)) as well as heterogeneous classifiers such as voting. The above ensemble classifiers were tested under various conditions include (i.e., with and without feature selection, with and without hyperparameter optimization). The optimized hyperparameters can be found in Appendix 3. Table 6 details all the experiments with ensemble classifiers. Note that stratified tenfold cross-validation was used to train and evaluate all models.

From Table 6, we can observe the following. Standard ensemble classifiers without feature optimization and hyperparameter optimization provide the worst performance RF (ACC = 0.606, AUC = 0.611), XGBoost (ACC = 0.621, AUC = 0.650), Bagging (ACC = 0.608, AUC = 0.631). Overall performance improved when using feature optimization (NSGA II) by (2–6)%, giving performance of RF (ACC = 0.642, AUC = 0.650), XGBoost (ACC = 0.822, AUC = 0.836), and Bagging (ACC = 0.827, AUC = 0.852). To improve classifier performance, we performed hyperparameter optimization using both BO and GSO. The best performance was obtained after tuning the bagging algorithm with the BO technique to achieve (ACC = 0.844, AUC = 0.853). Performance of the proposed ensemble DL model was compared with the best results from the standard ensembles. Figures 12, 13 illustrates that the proposed framework outperforms all other classifiers.

Table 4 Results of the ensemble classifiers

Classifier	MC	FO	CO	CV accuracy	ACC	Sn	Sp	AUC
Ensemble DNN	LR	–	–	0.797 ± 0.001	0.727	0.766	0.715	0.781
		NSGA-II	–	0.872 ± 0.055	0.865	0.812	0.801	0.851
		NSGA-II	GS	0.891 ± 0.032	0.872	0.882	0.831	0.863
		NSGA-II	BO	0.880 ± 0.009	0.880	0.861	0.811	0.898
	NN	–	–	0.808 ± 0100	0.781	0.749	0.713	0.752
		NSGA-II	–	0.870 ± 0.003	0.882	0.872	0.826	0.865
		NSGA-II	GS	0.916 ± 0.0282	0.890	0.909	0.821	0.886
		NSGA-II	BO	0.901 ± 0.0431	0.913	0.921	0.832	0.906

MC: Meta classifier, FO: Feature optimization, CO: Classifier optimization technique

Table 5 Comparison with scoring systems

Classifier or score	CV accuracy	ACC	Sp	Sn	AUC
MEWS	–	0.671	0.662	0.631	0.670
SOFA	–	0.740	0.705	0.777	0.78
Best single classifier	0.816 ± 0.0980	0.802	0.783	0.753	0.801
Best performance of ensemble classifier	0.901 ± 0.0431	0.913	0.921	0.832	0.906

Table 6 Standard ensemble classifiers results

Classifier	FO	CO	CV accuracy	ACC	Se	Sp	AUC
RF	–	–	0.618 ± 0.0035	0.606	0.717	0.501	0.611
	NSGA-II	–	0.650 ± 0.0021	0.642	0.686	0.607	0.632
	NSGA-II	GS	0.801 ± 0.0312	0.800	0.898	0.782	0.803
	NSGA-II	BO	0.851 ± 0.0088	0.843	0.857	0.781	0.821
XGBoost	–	–	0.632 ± 0.0009	0.621	0.754	0.501	0.650
	NSGA-II	–	0.667 ± 0.0119	0.653	0.857	0.800	0.762
	NSGA-II	GS	0.832 ± 0.0341	0.822	0.812	0.856	0.836
	NSGA-II	BO	0.866 ± 0.0441	0.851	0.892	0.842	0.864
Bagging	–	–	0.612 ± 0.0280	0.608	0.783	0.532	0.631
	NSGA-II	–	0.831 ± 0.0131	0.827	0.852	0.700	0.852
	NSGA-II	GS	0.851 ± 0.0080	0.843	0.856	0.832	0.862
	NSGA-II	BO	0.876 ± 0.00323	0.844	0.892	0.831	0.853

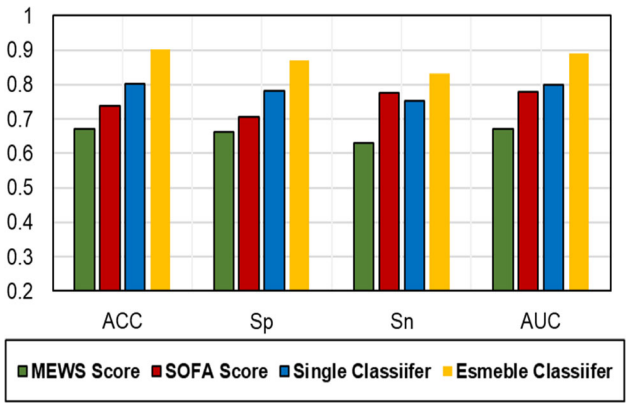

Fig. 11 Comparison with scoring systems

5.3.7 Comparison with the literature

As shown in Table 7, we compare our model with other state-of-the-art approaches from the literature in terms of performance and architecture. Note that we chose to only compare with studies that use the MIMIC dataset to provide a fair comparison. As shown in Table 7, most of the-state-the-art methods have followed the sepsis 3 definitions to determine sepsis in patients and depend on a small number of features to predict sepsis. Even though they achieved adequate results, these studies cannot be considered medically acceptable. They did not take into consideration the progression of various diseases which can result from sepsis. Medical experts usually depend on different measurements to diagnosis sepsis according to the patient's health status and what measurements are expected to be affected in case of sepsis—[57 and 61] particularly suffer from this limitation. Compared with [124], this study depends on a larger sample size of 4,915 patients. However, that study achieved a result of 0.750 in terms of AUC. This returns us to the issue of previous approaches depending on insight algorithm that are complex and do not consider changes in several important features. The same is true of [52], this study also depended on insights with a sample size of 1840 patients, resulting in an AUC of 0.781. He et al. [8] achieved sensitivity and specificity of 0.641 ± 0.022 and 0.844 ± 0.007, respectively, using an XGBoost ensemble classifier to predict sepsis in ICUs. That study used forty features from the MIMIC III dataset and used three LSTM models to extract deep features. However, that study has not proposed any ensemble models for the classification task. In [12, 29], authors used decision tree, and linear regression machine learning techniques for predicting sepsis resulting in AUCs of 0.890 and 0780, respectively, but these studies neglected the role of time in predicting sepsis and its effect in the progression of the patient's condition. As in our study, the authors in [50] used ensemble machine learning (i.e., XGBoost) to predict sepsis using only 6 features. Despite their impressive result in terms of an AUC = 0.880, this reliance on a small number of features means its predictions lack some credibility in the medical domain. One of the strengths of our model is its ability to predict sepsis using only the first

Table 7 Comparison with other models from the literature

References	Sample size	Data source	Features	Sepsis definition	Hours before onset	Algorithm	AUC
[124]	4915	MIMIC III	12	Sepsis 3	6 h	Insights	0.750
[12]	17,487	MIMIC II	8	Sepsis 3	–	Decision tree	0.890
[50]	2350	MIMIC III	6	SIRS criteria	–	XGB and MLP	0.880
[52]	1840	MIMIC II	–	Sepsis 3	7 h	*Insights*	0.880
[29]		MIMIC II	2	Sepsis 3	3 h	LR	0.780
[67]	6160	Data from Johns Hopkins PICU	126	Sepsis 3	–	XGBOOST	0.90
[65]	5154	MIMIC III	76	SOFA Score	6 h	CNN & RF	0.842
[57]	2970	MIMIC III	18	Sepsis 3		RF	0.81
[61]	40.336	MIMIC III	65	Sepsis 3	4,8,12	LSTM & CNN	0.89, 0.88, 0.87
[128]		MIMICIII	80	SIRS	6	DL	087–0.90
[66]	1588		106	Sepsis 3	8	XGBoost	0.89
Our model	4680	MIMIC III	22	SIRS Criteria	6–48 h	Ensemble DL	0.906

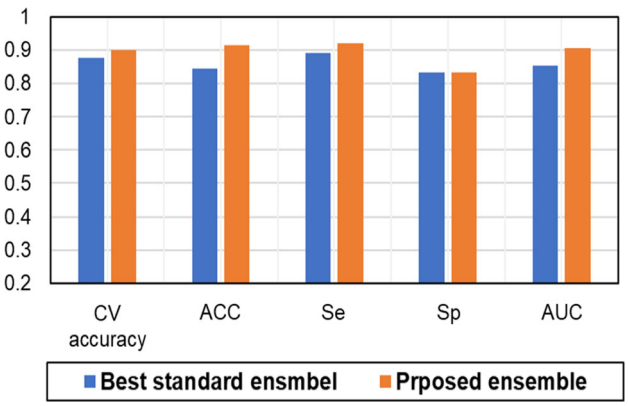

Fig. 12 Comparison between the best standard ensemble model and the proposed model

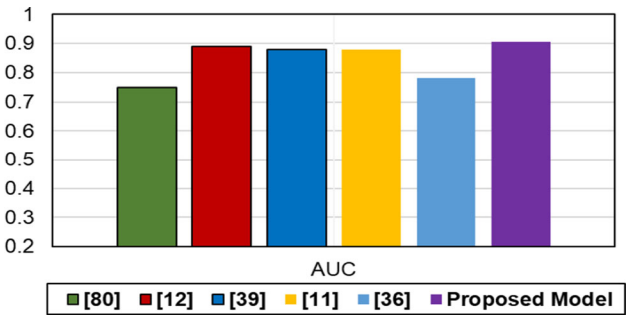

Fig. 13 Comparison with other models from the literature

six hours after patient admission. The results in the best model are not seen anywhere in the literature (AUC = 0.906).

5.4 Results of regression layer

This section investigates the use of machine learning and deep learning models to predict both the sepsis onset time and the blood pressure at that time. The novelty of this lies in designing a multitask deep learning model based on features extracted from the first six hours. We choose to predict blood pressure at the sepsis onset time, as it is the primary measurement that can be used to predict sepsis according to various studies [29, 32, 127]. *To the best of our knowledge, this is the first study that focuses on predicting sepsis onset time.* The period from which data is taken to make the prediction was chosen by a medical expert.

5.4.1 Regular machine learning model

In this section, we use machine learning methods including (i.e., LIR, Lasso, ridge, SGB, GB, RF, and DT) to predict both the onset time and the blood pressure at that time use traditional. The optimized hyperparameters can be found in Appendix 3. As shown in Table 8, the models were evaluated using mean absolute error (MAE) and Root mean square error (RMSE). First, f*or predicting the onset time*, we observed that SGD gives the highest error RMSE = 18.66 ± 1.98 and MAE = 17.22 ± 1.88, followed by linear regression which gives RMSE = 18.09 ± 1.65 and MAE = 17.88 ± 1.66. The best performance was obtained from RF with errors RMSE = 13.44 ± 2.88 and MAE = 13.89 ± 1.76. Figure 14a details the experiments for predicting onset time. Figure 14b shows the performance

Table 8 Machine learning regression models for both onset time and blood pressure

Predicting the onset time			
Model	The optimized hyper parameters	RMSE	MAE
LIR	fit_intercept = True, copy_X = True	18.09 ± 1.65	17.88 ± 1.66
RF	n_estimators = 100, max_depth = 2	13.88 ± 3.44	13.09 ± 3.32
Ridge	Alpha = 1.5	18.01 ± 2.22	18.66 ± 2.23
SGD	Alpha = 0.13, penalty = 'l2'	18.66 ± 1.98	17.22 ± 1.88
Lasso	Alpha = 1.0, normalize = False	17.61 ± 1.54	15.65 ± 1.64
GB	n_estimators = 100, max_depth = 2, learning_rate = 1.5	13.44 ± 2.88	13.89 ± 1.76
DT	max_depth = 3	16.22 ± 3.88	18.22 ± 2.89
Predicting the blood pressure in the onset time			
LIR	fit_intercept = True, normalize = True, copy_X = True	16.09 ± 2.55	15.78 ± 2.89
Lasso	Alpha = 1.6, normalize = False	19.88 ± 3.92	18.59 ± 4.02
Ridge	Alpha = 1.5	18.33 ± 3.12	17.17 ± 1.88
SGD	Alpha = 0.3, penalty = 'l2'	20.22 ± 1.66	19.67 ± 2.02
RF	n_estimators = 130, max_depth = 3	18.33 ± 3.02	18.05 ± 3.12
GB	n_estimators = 130, max_depth = 3, learning_rate = 1.5	16.01 ± 2.55	15.22 ± 2.55
DT	max_depth = 2	18.76 ± 2.81	17.96 ± 3.11

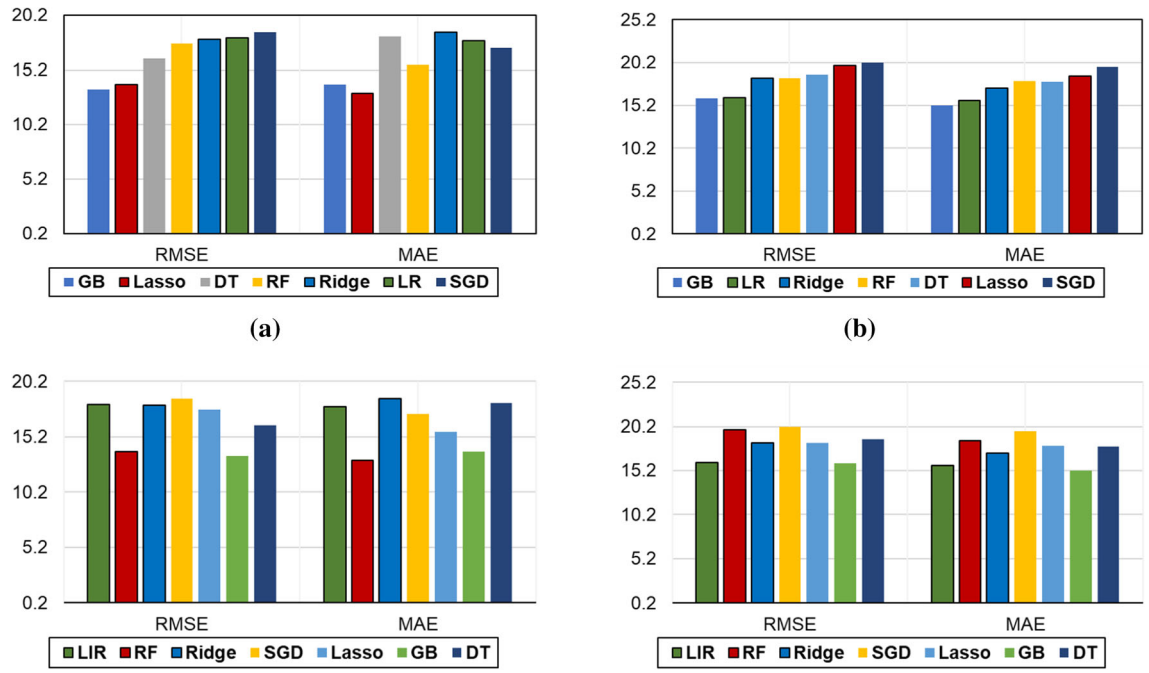

Fig. 14 (**a**) Prediction of the onset time, (**b**) predict the blood pressure at the sepsis onset time

for *the blood pressure prediction*, SGD gave the lowest performance RMSE = 20.22 ± 1.66, and MAE = 19.67 ± 2.02 followed by RF with error values RMSE = 18.33 ± 3.12 and MAE = 17.17 ± 1.88, the best performance was obtained with GB with RMSE = 16.01 ± 2.55 and MAE = 15.22 ± 2.55. The results show that there is an opportunity for improving the performance. Therefore, in the following section we utilize single task and multitask DL models to improve performance and develop more robust and confident models.

5.4.2 Results of the deep learning model

Multitask learning is a multi-objective problem where the overall optimization of the DL model can be improved. In this section we look at concurrently optimizing two regression tasks. The developed model tells medical experts the onset time and the predicted blood pressure at that time. To the best of our knowledge, this study is the first study that predicts the future onset of sepsis in a patient from the first 6 h of that patient's data. These

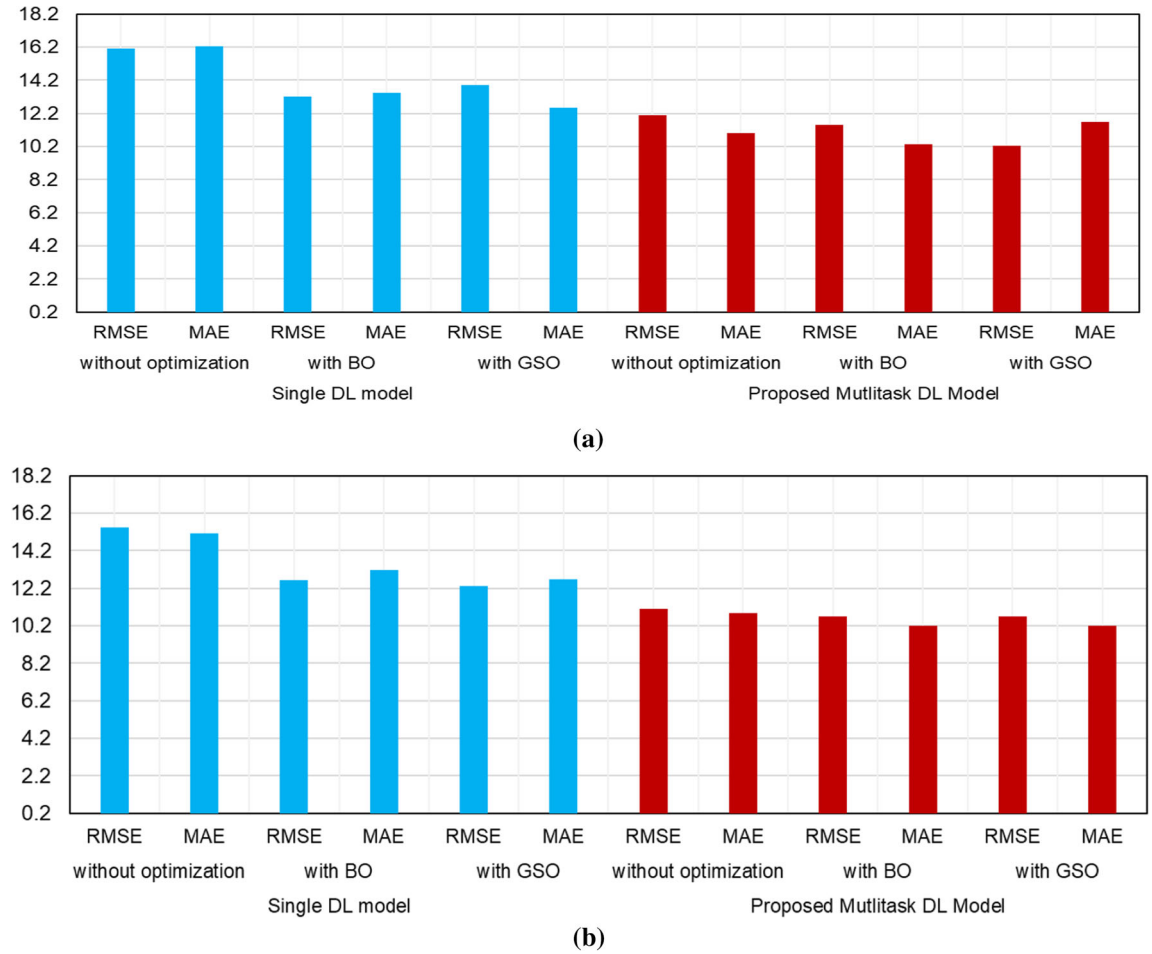

Fig. 15 (**a**) Predicting onset time using single and multitask models, (**b**) predicting blood pressure at onset time using single and multitask models

experiments followed the same procedure as the previous tests. First, we tested single task deep learning models. We used 20 hidden layers (chosen by trial and error) and used the ReLU activation function in the hidden layers. The model was fit using mean absolute error (MAE), Root mean square error (RMSE), and an Adam optimizer (stochastic gradient descent). The optimized hyperparameters can be found in Appendix 3. Using single task deep learning resulted in a prediction model with (RMSE = 16.11 ± 3.87) and (MAE = 16.23 ± 3.33) for predicting onset time, and (RMSE = 15.44 ± 2.55) (MAE = 15.13 ± 3.44) for predicting the blood pressure at that time. To enhance performance, we tuned the hyperparameters using GSO and BO. Tuning the classifier with BO enhanced the results by 2–4%, with (RMSE = 13.23 ± 2.11) and (MAE = 13.45 ± 1.88) for predicting onset time and (RMSE = 12.65 ± 3.12), (MAE = 13.17 ± 1.87) for predicting the blood pressure at that time. The best performance came after tuning with GSO achieving (RMSE = 13.88 ± 2.78) (MAE = 12.56 ± 2.02) for predicting onset time and (RMSE = 12.34 ± 1.66), (MAE = 12.67 ± 2.98) for predicting the blood pressure at that time. The first half of Fig. 15a shows the results of using the single task DL model to predict the onset time. Moreover, Fig. 15b shows the performance of predicting the blood pressure during the beginning of the onset time using single task DL.

Second, we utilized multitask modeling, which allows us to share important information between tasks. Theoretically, multitask modeling should improve the performance, make the model more stable and give more confidence in its predictions, the results we achieved with our multitask model back this up. Table 9 clarifies the performance improvement of the multitask model compared with the single task models. We utilized the multitask model to predict two metrics: the sepsis onset time (to a particular hour interval) and the blood pressure at that time. It is worth noting that using a multitask DL model achieved better performance than using single task DL models. We are also interested in tackling the challenges that come with multitask models. For example, in our model with two regression tasks, the model suffered from overfitting with different rates. To solve this issue, we

Table 9 Performance of single task and multitask regression models for both onset time and blood pressure

Predicting the onset time				Predicting the blood pressure	
Model	Optimization	RMSE	MAE	RMSE	MAE
Single deep learning model	–	16.11 ± 3.87	16.23 ± 3.33	15.44 ± 2.55	15.13 ± 3.44
	BO	13.23 ± 2.11	13.45 ± 1.88	12.65 ± 3.12	13.17 ± 1.87
	GSO	13.88 ± 2.78	12.56 ± 2.02	12.34 ± 1.66	12.67 ± 2.98
Predicting the onset time				Predicting the blood pressure	
Model	optimization	RMSE	MAE	RMSE	MAE
Multitask deep learning model	–	12.09 ± 1.11	11.03 ± 2.31	11.11 ± 2.62	10.88 ± 2.89
	BO	11.52 ± 2.31	10.32 ± 1.83	10.69 ± 2.31	10.19 ± 1.90
	GSO	10.26 ± 1.66	11.67 ± 1.94	9.22 ± 1.31	9.67 ± 1.63

propose the use of heuristics, including the use of the early stopping technique by defining a specific number of epochs for each task based on single task training. We also propose the use of single task loss weighting, which may also reduce the problem. As we can see in Table 9, the multitask model achieved reduced errors for both RMSE and MAE with values ranging from (0 to 3) with (RMSE = 12.09 ± 1.11) (MAE = 11.03 ± 2.31) for predicting onset time and (RMSE = 11.11 ± 2.62), (MAE = 10.88 ± 2.89) for predicting the blood pressure at that time. As expected, using multitask learning for related tasks improves overall performance, resulting in a more robust model. Considering a multi-objective function helps us in the optimization of a single task model, rather than being sensitive to the performance of every single task. The second half of Fig. 15a shows the results of using a multitask DL model to predict the onset time. We also observed that tuning the hyperparameters contributed to improving the performance of the regressor. Using BO achieved a performance of RMSE = 10.69 ± 2.31 and MAE = 10.19 ± 1.90. The lowest error was obtained when using the GSO technique achieving RMSE = 9.22 ± 1.31 and MAE = 9.67 ± 1.63.

6 Study limitations

Although the proposed model gives us a promising tool for sepsis prediction, it still has many limitations that need further attention. *First*, because the MIMIC III dataset is extracted from one institution, we cannot guarantee the generalization of our results. In future studies, we will explore other datasets. *Second*, we depend on the ICD 9 codes gold standard to define sepsis, this may result in failing to detect all sepsis patients in the dataset. *Third*, the imputation process in which we average all measurements for each hour period may lead to the loss of some temporal values which could negatively affect model performance. Therefore, we intend to work on better imputation techniques to capture this missing data. Fourth, the sequence of lab test results mainly depends on physician requests. Accordingly, the gold standard is highly subjective. Therefore, finding a consistent gold standard definition is an important goal during future exploration in this area. Fifth, summarizing time series data and working with feedfoward neural networks may mean discarding many temporal features in those multivariate series. Utilizing other deep learning models such as LSTM and CNN are expected to improve the performance of both classification and regression models. These limitations will be addressed in our future studies.

7 Conclusion

This paper proposed a multitask multilayer model for classification and regression tasks. The model was applied to predict sepsis, sepsis onset time, and blood pressure at that time. First, data were preprocessed for irregular time sampling, outlier detection, and imbalanced classes. Second, NSGA-II was used for feature selection. NSGA II, a feature subset selection algorithm, was combined with an ANN, a learning algorithm, to discover the optimal set of features that will minimize errors. Third, the optimal feature set extracted during the second phase was used to build an ensemble neural network classifier. This system was tested with data from 4500 patients in its task to predict sepsis (6–48) hours before the onset time. This makes up the model's first layer. Data from 2350 sepsis patients were used for multitask learning to predict patient's vital signs. This makes up the model's second layer. Our proposed deep learning model performed better than several scores, i.e., SOFA and MEWS, that are traditionally used to identify sepsis. The proposed classification model achieved an accuracy of 0.913, a specificity of 0.921, a recall of 0.832, and an AUC of 0.906. Proposed multitask regression model achieved an RMSE of 10.26 and 9.22 for predicting the onset time and the blood pressure at that time, respectively. In the future, we will test our model with

other real ICU data sets. We will explore the role other deep learning models could play, including LSTM and CNN, when dealing with time-series data. Finally, we will provide domain experts with clear justifications for decisions coming from our model.

Appendix 1

See Tables 10, 11.

Table 10 List of abbreviations

Abbreviation	Term
ADASYN	Adaptive synthetic sampling
ANN	Artificial neural network
AUC	Area under the roc curve
BO	Bayesian optimization
CHMM	Coupled hidden Markov models
CNN	Convolutional neural network
CO	Classifier optimization
CV	Cross-validation
DIC	Disseminated intravascular coagulation
DL	Deep learning
DNN	Deep neural network
DT	Decision tree
EHR	Electronic health record
ESICM	European society of intensive care medicine
FO	Feature optimization
FS	Feature extraction
GB	Gradient boosting regression
GSO	Both grid search optimization
ICU	Intensive care unit
IG	Information gain
LASSO	Lasso regression
LOS	Length of stay
LSTM	Long short-term model
LR	Logistic regression
LIR	Linear regression
CLS	Classifier Learning rate
MAE	Mean absolute error
MC	Meta classifier
MEWS	Modified early warning score
MICRO- GA	Micro genetic algorithm
MIMIC III	Medical information mart from the intensive care MIMIC-III
ML	Machine learning
MLP	Multilayer perceptron
MOGA	Multi-objective genetic algorithm
MTL	Multitask learning
NSGA II	Non-dominated sorting genetic algorithm II
RELU	Rectified linear unit
RF	Random forest
RMSE	Root mean square error
RNN	Recurrent neural network
SCCM	Society of critical care medicine
SIRS	Systemic inflammatory response syndrome
SMOTE	Synthetic minority oversampling technique
SOFA	Sequential organ failure assessment
SVM	Support vector machine
WBC	White blood cells

Table 10 (continued)

Abbreviation	Term
XGBOOST	Extreme gradient boosting
SICU	Surgical intensive care unit
MICU	Medical intensive care unit
CRSU	Cardiac surgery recovery unit

Table 11 Features used for Sepsis

F. No	Item_id	Feature name	T. Name	Type	Normal range	UoM	Max	mean	min
	Demographic								
1	–	Age	Patients	N	–	Y	90	35	15
2	–	Gender	Patients	C	–	–	–	–	-
3	–	Weight	ICU stays	N	–	kg	160	73	47
4	–	BMI	Patient & ICU	N	–	kg/m2	50	25	15
5	50,863	Alkaline phosphatase	Chartevents	N	30–250	U	300	151.23	20
6	50,862	Albumin	Chartevents	N	3.5 – 5	g	5.6	3.5–5	.8
7	50,910	Creatine kinase (CK)	Lab events	N	15- 105	U/L	2238	85	3
8	1525	Creatinine	Chartevents	N	7 to 1.3	Ml/dl	647	12	0
9	675	Blood Urea Nitrogen (BUN)	Chartevents	N	10–20	Ml/dl	270	0	32.88
10	411	Heart rate	Chartevents	N	60–100	p/m	300	60	40
11	2381	Respiratory rate	Chartevents	N	14- 40	B/m	60	18	6
12	646	SpO4	Chartevents	N	94–100%	%	40	96	100
13	616	Respiratory effort	Chartevents	C	–	–	–	–	-
14	677	Temperature C (calc)	Chartevents	N	37.5	mmHg	41..5	0	30
15	440,054	Arterial blood pressure means	Chartevents	N	100–140	mmHg	95.1	90	141.8
16	440,050	Arterial blood pressure systolic	Chartevents	N	140	mmHg	171	-135	81.44
17	440,051	Arterial blood pressure diastolic	Chartevents	N	80	mmHg	150	80	30
18	780	Arterial pH	Chartevents	N	7.35- 7.45	mmHg	7.87	6	6.80
19	778	Arterial PaCO4	Chartevents	N	35–45	mm Hg	100	54	10
20	779	Arterial PaO2	Chartevents	N	88–100	mmHg	500	340	40
21	444,848	Arterial base excess	Chartevents	N	-2 + 2	mEq/L	4	+ 1	-10
22	198	GSC total	Chartevents	N	15	–	15	–	3
22	40,069	Urine out void	outputevents	N	.3- .5 ml for kg per H	Ml/dl	–	–	-
23	40,086	Lactate	Chartevents	N	4–4	mmol/L	5	3	4
24	445,664	Glucose level	Chartevents	N	80–140	Ml/dl	40	110	500
25	447,457	Platelet count	Chartevents	N	150–400(000)	N	140	438	400
26	813	Hematocrit	Chartevents	N	39- 44 M,35–45 F	%	36		64
27	440,448	Hemoglobin	Chartevents	N	13–17 M,14–16 F	g/dl	40	14	1
28	1146	Art.pH	Chartevents	N	7.8.7.44	–	7.8	7	0
29	440,546	WBC	Chartevents	N	4000–11,000	N	4	7	11
30	490	PAO4	Chartevents	N	88–100	mmHg	160	93	60
31	447,466	partial thromboplastin time (PTT)	Chartevents	N	40–70	second	89	55	14
32	447,073	Anion gap	Chartevents	N	3–11	L	0	8	54
33	861	WBC	Chartevents	N	(4–11,0000)	num	11	560	3
34	447,443	HCO3 (serum)	Chartevents	N	44–46	N	40	30	0
35	227,467	INR	Chartevents	N	2.0–3.0	–	1.1	2	> 3
36	43,365	urine output/kg/hr	Chartevents	N	0.3–0.5	ML	0.1	0.4	0.8
37	3801	SGOT	Chartevents	N	8–48	Unit/L	8	45	50
38	3802	SGPT	Chartevents	N	7–65	Unit/L	5	42	60
39	942	Blood culture	Chartevents	–	–	–	–	–	–

Appendix 2

See Tables 12, 13, 14.

Table 12 Sepsis-related organ failure assessment scoring system

SOFA Score					
Features	0	1	2	3	4
Pao_2/fio_2 mm Hg	≥ 400	< 400	< 300	< 200	< 100
Platelets $\times 10^3/\mu L$	≥ 150	< 150	< 100	< 50	< 20
Bilirbubin mg/dL	< 1.2	1.2–1.9	2.0- 5.9	6.0–11.9	> 12.0
Cardiovascular	MAP $\geq$ 70 mm Hg	MAP $<$ 70 mm Hg	Dopamine < 5	Dopamine (5.1- 15)	12.0
Glasgow Coma Score	15	13–14	10–12	6–9	< 6
Creatinine mg/dL	< 1.2	1.2–1.9	2.0–3.4	3.5–4.9	> 5.0
Urine output, mL/dl				< 500	< 200

Table 13 qSOFA Score

qSOFA (Quick SOFA)	Points
Criteria Points Respiratory rate $\geq$ 22/min 1 1	1
Change in mental status	1
Systolic blood pressure $\leq$ 100 mmHg	1

Table 14 Modified Early Warning Score

MEWS Score							
Features	3	2	1	0	1	2	3
Heart rate	< 70	71–80	81–100	101–199		> 200	
Systolic blood pressure		< 40	4050	51–100	101–110	111–129	> 130
Respiratory rate (RR)		< 9		9–14	15–20	21–29	> 30
Temprature		< 35		35–38.4	12.0	$> 38/5$	

Appendix 3

See Table 15.

Table 15 Hyperparameters for NSGA-II, ensemble model, regular classifiers, regular regressors, and deep learning regressor

NSGA II hyperparameters	Value
Min number of features	10
Population size	100
Max number of generations	80
Selection Schema	Non dominant sorting
Maximal fitness	Infinity
P Inilaize	0.5
P mutation	-1.0
P cross over	0.5

Table 15 (continued)

NSGA II hyperparameters	Value
Crossover Type	Uniform
Normalize weights	Yes

DL regression model	Value
Regularization	L2 = 0.2
Dropout	0.1
Batch size	128
Activation function in hidden layers	ReLU
Number of epochs	100
Batch size	128
Number of hidden layers	15
Activation function in output layer	ReLU
Optimizer used	ADAM
Loss function	Mean squared error

Model	Hyperparameters for onset time prediction
LR	fit_intercept = True, copy_X = True
RF	n_estimators = 100, max_depth = 2
Ridge	Alpha = 1.5
SGD	Alpha = 0.13, penalty = 'l2'
Lasso	Alpha = 1.0, normalize = False
GB	n_estimators = 100, max_depth = 2, learning_rate = 1.5
DT	max_depth = 3
Model	Hyperparameters for blood pressure prediction
LR	fit_intercept = True, normalize = True, copy_X = True
Lasso	Alpha = 1.6, normalize = False
Ridge	Alpha = 1.5
SGD	Alpha = 0.3, penalty = 'l2'
RF	n_estimators = 130, max_depth = 3
GB	n_estimators = 130, max_depth = 3, learning_rate = 1.5
DT	max_depth = 2

Ensemble model hyperparameters	Value
Number of Deep learning models	4 models
Input layer	263 unit
Number of layers	20
Regularization	L2 = 0.1
Dropout	0.1
Batch size	128
Activation function in hidden layers	ReLU
Number of epochs	100
Batch size	300
Number of hidden layers	15
Activation function in output layer	Sigmoid
Optimizer used	ADAM
Meta classifiers	NN, LR

Funding This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ICT Creative Consilience Program(IITP-2021-2020-0-01821) supervised by the IITP(Institute for Information & communications Technology Planning & Evaluation), and the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. 2021R1A2C1011198).

Declarations

Conflict of interest The authors declare that they have no conflicts of interest.

References

1. Gaieski DF, Edwards JM, Kallan MJ, Carr BG (2013) Benchmarking the incidence and mortality of severe sepsis in the United States. Crit Care Med 41(5):1167–1174
2. Fein AM, Balk RA, Knaus WA, Schein RMH, Dellinger RP and Sibbald W (1991) Definitions for sepsis and organ failure and guidelines for the use of innovative therapies in sepsis
3. Levy MM et al. (2003) International sepsis definitions conference, pp. 530–538
4. Challenges I and Standard G (2015) The third international consensus definitions for sepsis and septic shock (Sepsis-3), 18(2):162–164
5. Outcomes M (2015) Mortality related to severe sepsis and septic shock among critically Ill patients in Australia and New Zealand, 2000–2012
6. Daviaud F et al. (2015) Timing and causes of death in septic shock. Ann Intens Care
7. Layeghian Javan S, Sepehri MM, Layeghian Javan M, Khatibi T (2019) An intelligent warning model for early prediction of cardiac arrest in sepsis patients. Comput Methods Progr Biomed 178:47–58
8. He Z, Du L, Zhang P, Zhao R, Chen X, and Fang Z (2020) Early sepsis prediction using ensemble learning with deep features and artificial features extracted from clinical electronic health records. pp. 1337–1342
9. Opal SM, Rubenfeld GD, Van Der Poll T, Vincent J, and Angus DC (2016) The third international consensus definitions for sepsis and septic shock (Sepsis-3), 315(8):801–810
10. Rhodes A et al. (2017) Surviving sepsis campaign: international guidelines for management of sepsis and septic shock, 2016, 43(3). Springer: Berlin Heidelberg
11. Calvert JS et al (2016) A computational approach to early sepsis detection. Comput Biol Med 74:69–73
12. Mao Q et al (2018) Multicentre validation of a sepsis prediction algorithm using only vital sign data in the emergency department, general ward and ICU. BMJ Open 8(1):1–11
13. Desautels T et al. (2016) Prediction of sepsis in the intensive care unit with minimal electronic health record data: a machine learning approach corresponding author 4:1–15
14. Verdonk F, Blet A, and Mebazaa A (2017) The new sepsis definition: limitations and contribution to research and diagnosis of sepsis. Curr Opin Anesthesiol, 30(2)
15. Buehler SS et al (2016) Effectiveness of practices to increase timeliness of providing targeted therapy for inpatients with bloodstream infections: a laboratory medicine best practices systematic review and meta-analysis. Clin Microbiol Rev 29(1):59–103
16. El-Sappagh S, Abuhmed T, Riazul Islam SM, and Kwak KS (2020) Multimodal multitask deep learning model for alzheimer's disease progression detection based on time series data. Neurocomputing
17. Abuhmed T, El-sappagh S, Alonso JM (2021) Robust hybrid deep learning models for Alzheimer's progression detection. Knowl Based Syst 213:106688
18. Si Y and Roberts K, Deep patient representation of clinical notes via multi-task learning for mortality prediction, pp. 1–10
19. Garg A, Mago V (2021) Role of machine learning in medical research: a survey. Comput Sci Rev 40:100370
20. Kam HJ, Kim HY (2017) Learning representations for the early detection of sepsis with deep neural networks. Comput Biol Med 89:248–255
21. Chen C-Y, and Huang J-J (2020) Integration of genetic algorithms and neural networks for the formation of the classifier of the hierarchical choquet integral. Inf Sci (Ny)
22. Fang H, Wang Q, Tu YC, Horstemeyer MF (2008) An efficient non-dominated sorting method for evolutionary algorithms. Evol Comput 16(3):355–384
23. Hamdani TM, Won JM, Alimi AM, and Karray F (2007) Multi-objective feature selection with NSGA II, Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinformatics), 4431 LNCS, no. Part 1, pp. 240–247
24. Yusoff Y, Ngadiman MS, Zain AM (2011) Overview of NSGA-II for optimizing machining process parameters. Procedia Eng 15:3978–3983
25. De Silva N, Ranasinghe M, De Silva CR, and Thurairajah N (2012) Architecture of ensemble neural networks for risk analysis. In: 48th ASC Ann Int Conf Proc, no. April, pp. 1–9
26. Hansen LK, Salamon P (1990) Neural network ensembles. IEEE Trans Pattern Anal Mach Intell 12(10):993–1001
27. Perrone MP and Cooper LN (1995) When networks disagree: Ensemble methods for hybrid neural networks, no. February, pp. 342–358
28. Maclin R (2016) Popular ensemble methods: an empirical study popular ensemble methods: an empirical study, 11:169–198
29. Shashikumar SP et al (2017) Early sepsis detection in critical care patients using multiscale blood pressure and heart rate dynamics. J Electrocardiol 50(6):739–743
30. Hug ATRCW, Clifford GD (2012) Clinician blood pressure documentation of stable intensive care patients: an intelligent archiving agent has a higher association with future hypotension, 39(5):1006–1014
31. Lehman LH, Mark RG, and Nemati S (2016) A model-based machine learning approach to probing autonomic regulation from nonstationary vital-sign time series, 2194(c):1–11
32. Singer M et al (2016) The third international consensus definitions for sepsis and septic shock (sepsis-3). JAMA 315(8):801–810
33. Kaukonen K-M, Bailey M, Bellomo R (2015) Systemic inflammatory response syndrome criteria for severe sepsis. The New Engl J Med 373(9):881
34. Churpek MM, Zadravecz FJ, Winslow C, Howell MD, Edelson DP (2015) Incidence and prognostic value of the systemic inflammatory response syndrome and organ dysfunctions in ward patients. Am J Respir Crit Care Med 192(8):958–964
35. Whippy A et al (2011) Kaiser Permanente's performance improvement system, part 3: multisite improvements in care for patients with sepsis. Jt Comm J Qual patient Saf 37(11):483–493
36. Shankar-Hari M et al (2016) Developing a newdefinition and assessing newclinical criteria for Septic shock: for the third international consensus definitions for sepsis and septic shock (sepsis-3). JAMA 315(8):775–787
37. Seymour CW et al (2016) Assessment of clinical criteria for sepsis for the third international consensus definitions for sepsis and septic shock (sepsis-3). JAMA 315(8):762–774

38. Marik PE, Taeb AM (2017) SIRS, qSOFA and new sepsis definition. J Thorac Dis 9(4):943–945
39. Schinkel M, Paranjape K, Nannan Panday RS, Skyttberg N, Nanayakkara PWB (2019) Clinical applications of artificial intelligence in sepsis: a narrative review. Comput Biol Med 115:103488
40. Darwiche A, Mukherjee S (2018) Machine learning methods for septic shock prediction. ACM Int Conf Proc Ser 1051:104–110
41. Arabi Y, Al Shirawi N, Memish Z, Venkatesh S, Al-Shimemeri A (2003) Assessment of six mortality prediction models in patients admitted with severe sepsis and septic shock to the intensive care unit: a prospective cohort study. Crit Care 7(5):R116–R122
42. Liu R et al (2019) Data-driven discovery of a novel sepsis pre-shock state predicts impending septic shock in the ICU. Sci Rep 9(1):1–9
43. Ghosh S, Li J, Cao L, Ramamohanarao K (2017) Septic shock prediction for ICU patients via coupled HMM walking on sequential contrast patterns. J Biomed Inform 66:19–31
44. Scherpf M, Gräßer F, Malberg H, Zaunseder S (2019) Predicting sepsis with a recurrent neural network using the MIMIC III database. Comput Biol Med 113:103395
45. Fagerström J, Bång M, Wilhelms D, Chew MS (2019) LiSep LSTM: a machine learning algorithm for early detection of septic shock. Sci Rep 9(1):1–8
46. Song W, Jung SY, Baek H, Choi CW, Jung YH, Yoo S (2020) A predictive model based on machine learning for the early detection of late-onset neonatal sepsis: development and observational study. JMIR Med Inform 8(7):e15965
47. Yao R et al (2020) A machine learning-based prediction of hospital mortality in patients with postoperative sepsis. Front Med 7:445
48. Jones CN et al (2014) Spontaneous neutrophil migration patterns during sepsis after major burns. PLoS ONE 9(1):e114509
49. Lukaszewski RA et al. (2008) Presymptomatic prediction of sepsis in intensive care unit patients, 15(7):1089–1094
50. Mitra A and Ashraf K (2018) Sepsis prediction and vital signs ranking in intensive care unit patients
51. Calvert J et al (2017) Cost and mortality impact of an algorithm-driven sepsis prediction system. J Med Econ 20(6):646–651
52. Desautels T et al (2016) Prediction of sepsis in the intensive care unit with minimal electronic health record data: a machine learning approach. JMIR Med Inform 4(3):e28
53. Johnson LSAEW, Pollard TJ (2016) Data Descriptor: MIMIC-III, a freely accessible critical care database. Thromb Haemost 76(2):258–262
54. Scherpf M, Zaunseder S, and Dortmund A (2019) Predicting sepsis with a recurrent neural network using the MIMIC III database Predicting sepsis with a recurrent neural network using the MIMIC III Database, no. August
55. Stanski NL, Wong HR (2020) Prognostic and predictive enrichment in sepsis. Nat Rev Nephrol 16(1):20–31
56. Pettinati MJ, Chen G, Rajput KS, and Selvaraj N (2020) Practical machine learning-based sepsis prediction. In: 2020 42nd annual international conference of the IEEE engineering in medicine biology society (EMBC), pp. 4986–4991
57. Chen M, Hernández A (2021) Towards an explainable model for sepsis detection based on sensitivity analysis. IRBM 1:1–12
58. Schellenberger S, Shi K, Wiedemann JP, Lurz F, and Weigel R, An ensemble LSTM architecture for clinical sepsis detection, pp. 2–5
59. Zabihi M, Kiranyaz S, and Gabbouj M (2019) Sepsis prediction in intensive care unit using ensemble of XGboost models, in 2019 computing in cardiology (CinC), 1– 4
60. Zhang D et al (2021) An interpretable deep-learning model for early prediction of sepsis in the emergency department. Patterns 2(2):100196
61. Rafiei A, Rezaee A, Hajati F, Gheisari S, Golzan M (2021) SSP: early prediction of sepsis using fully connected LSTM-CNN model. Comput Biol Med 128:104110
62. Tsang G and Xie X (2021) Deep learning based sepsis intervention: the modelling and prediction of severe sepsis onset. In: 2020 25th international conference on pattern recognition (ICPR), pp. 8671–8678
63. Al-mualemi BY and Lu LU (2021) A deep learning-based sepsis estimation scheme, 9
64. Goh KH et al (2021) Artificial intelligence in sepsis early prediction and diagnosis using unstructured data in healthcare. Nat Commun 12(1):711
65. Aşuroğlu T, Oğul H (2021) A deep learning approach for sepsis monitoring via severity score estimation. Comput Methods Prog Biomed 198:105816
66. Yuan K-C et al (2020) The development an artificial intelligence algorithm for early sepsis diagnosis in the intensive care unit. Int J Med Inform 141:104176
67. Liu R, Greenstein JL, Fackler JC, Bergmann J, Bembea MM, and Winslow RL (2021) Prediction of impending septic shock in children with sepsis. Crit Care Explor, 3(6)
68. Ngufor C, Upadhyaya S, Murphree D, Kor D, and Pathak J (2015) Multi-task learning with selective cross-task transfer for predicting bleeding and other important patient outcomes. In: 2015 IEEE International conference on data science and advanced analytics (DSAA), pp. 1–8
69. Giacobbe DR et al. (2021) Early detection of sepsis with machine learning techniques: a brief clinical perspective, 8
70. Fleuren LM et al (2020) Machine learning for the prediction of sepsis: a systematic review and meta-analysis of diagnostic test accuracy. Intens Care Med 46(3):383–400
71. Vellido A, Ribas V, Morales C, Ruiz Sanmartín A, Ruiz Rodríguez JC (2018) Machine learning in critical care: state-of-the-art and a sepsis case study. Biomed Eng Online 17(S1):1–18
72. Le S et al (2019) Pediatric severe sepsis prediction using machine learning. Front Pediatr 7:1–8
73. Kausch SL, Moorman JR, Lake DE, Keim-Malpass J (2021) Physiological machine learning models for prediction of sepsis in hospitalized adults: an integrative review. Intens Crit Care Nurs 65:103035
74. Hassan N et al (2021) Preventing sepsis how can artificial intelligence inform the clinical decision-making process a systematic review. Int J Med Inform 150:104457
75. Xie Y et al (2021) A prediction model of sepsis - associated acute kidney injury based on antithrombin III. Clin Exp Med 21(1):89–100
76. Calvert J et al (2016) Using electronic health record collected clinical variables to predict medical intensive care unit mortality. Ann Med Surg 11:52–57
77. Fullerton JN, Price CL, Silvey NE, Brace SJ, Perkins GD (2012) Is the Modified early warning score (MEWS) superior to clinician judgement in detecting critical illness in the pre-hospital environment? Resuscitation 83(5):557–562
78. Chandrashekar G, Sahin F (2014) A survey on feature selection methods. Comput Electr Eng 40(1):16–28
79. Bucholc M et al (2019) A practical computerized decision support system for predicting the severity of Alzheimer's disease of an individual. Exp Syst Appl 130:157–171
80. Jain D, Singh V (2018) Feature selection and classification systems for chronic disease prediction: a review. Egypt Inform J 19(3):179–189
81. Kohavi R, John GH (1997) Wrappers for feature subset selection. Artif Intell 97(1–2):273–324

82. Sharma M and Kaur P (2021) A comprehensive analysis of nature-inspired meta-heuristic techniques for feature selection problem. Arch Comput Methods Eng, 28(3)
83. Nguyen BH, Xue B, Zhang M (2020) A survey on swarm intelligence approaches to feature selection in data mining. Swarm Evol Comput 54:100663
84. Vieira SM, Mendonça LF, Farinha GJ, Sousa JMC (2013) Modified binary PSO for feature selection using SVM applied to mortality prediction of septic patients. Appl Soft Comput 13(8):3494–3504
85. Li Y, Li T, Liu H (2017) Recent advances in feature selection and its applications. Knowl Inf Syst 53(3):551–577
86. Huang B, Buckley B, Kechadi T-M (2010) Multi-objective feature selection by using NSGA-II for customer churn prediction in telecommunications. Exp Syst Appl 37(5):3638–3646
87. Kurnaz S, Mohammed MS, and Mohammed SJ (2020) A high efficiency thyroid disorders prediction system with non-dominated sorting genetic algorithm NSGA-II as a feature selection algorithm. In: 2020 International Conference for Emerging Technology (INCET), pp. 1–6 s
88. Zhang Y, Gong D, Cheng J (2015) Multi-objective particle swarm optimization approach for cost-based feature selection in classification. IEEE/ACM Trans Comput Biol Bioinforma 14(1):64–75
89. Khan A, Baig AR (2015) Multi-objective feature subset selection using non-dominated sorting genetic algorithm. J Appl Res Technol 13(1):145–159
90. Zangooei MH, Habibi J, Alizadehsani R (2014) Disease diagnosis with a hybrid method SVR using NSGA-II. Neurocomputing 136:14–29
91. Soui M, Mansouri N, Alhamad R, Kessentini M and Ghedira K (2021) NSGA-II as feature selection technique and AdaBoost classifier for COVID-19 prediction using patient's symptoms. Nonlinear Dyn. pp. 1–23
92. Soyel H, Tekguc U, Demirel H (2011) Application of NSGA-II to feature selection for facial expression recognition. Comput Electr Eng 37(6):1232–1240
93. Zitzler E, Deb K, Thiele L (2000) Comparison of multiobjective evolutionary algorithms: empirical results. Evol Comput 8(2):173–195
94. Salmanpour MR et al (2019) Optimized machine learning methods for prediction of cognitive outcome in Parkinson's disease. Comput Biol Med 111:103347
95. Türkşen Ö, Vieira SM, Madeira JFA, Apaydin A, and Sousa JMC (2013) Comparison of multi-objective algorithms applied to feature selection," in Towards advanced data analysis by combining soft computing and statistics, Springer, pp. 359–375
96. Hojjati A, Monadi M, Faridhosseini A, Mohammadi M (2018) Application and comparison of NSGA-II and MOPSO in multi-objective optimization of water resources systems. J Hydrol Hydromech 66(3):323–329
97. Wang R (2016) An improved nondominated sorting genetic algorithm for multiobjective problem. Math Probl Eng, 2016
98. Khan A and Baig A, Multi-objective feature subset sorting genetic algorithm selection using. J Appl Res Technol, 13(1):145–159
99. Wang Q, Wang L, Huang W, Wang Z, Liu S, and Savić DA (2019) Parameterization of NSGA-II for the optimal design of water distribution systems. Water (Switzerland), 11(5)
100. Zhang C and Ma X (2015) NSGA-II algorithm with a local search strategy for multiobjective optimal design of dry-type air-core reactor. Math Probl Eng, 2015
101. Indexed S, Bachri OS, Program AI, Hatta M, and Nurhayati OD (2017) Feature selection based on chi square in artificial neural network to predict the accuracy of student, 8(8):731–739
102. Nti IK, Adekoya AF, and Weyori BA (2020) A comprehensive evaluation of ensemble learning for stock-market prediction. J Big Data
103. Zhang Y and Yang Q (2017) A survey on multi-task learning, pp. 1–20
104. Ruder S (2017) An overview of multi-task learning in deep neural networks
105. Chen W, Long G, Yao L, and Sheng QZ (2019) AMRNN: attended multi-task recurrent neural networks for dynamic illness severity prediction
106. Harutyunyan H, Khachatrian H, Kale DC, Ver Steeg G, Galstyan A (2019) Multitask learning and benchmarking with clinical time series data. Sci Data 6(1):96
107. Razavian N, Marcus J, and Sontag D, Lab Tests, pp. 1–27
108. Joenssen DW and Bankhofer U (2015) Hot deck methods for imputing missing data hot deck methods for imputing missing data the effects of limiting donor usage, no. July 2012
109. Rahman MM, Davis DN (2013) Addressing the class imbalance problem in medical datasets. Int J Mach Learn Comput 2013:224–228
110. Li D-C, Liu C-W, Hu SC (2010) A learning method for the class imbalance problem with medical data sets. Comput Biol Med 40(5):509–518
111. Moreo A and Esuli A (2016) Distributional random oversampling for imbalanced text classification pp. 805–808
112. Chawla NV, Bowyer KW, Hall LO, and Kegelmeyer WP (2002) SMOTE: synthetic minority over-sampling technique
113. Li S (2015) ADASYN: adaptive synthetic sampling approach for imbalanced, no. July 2008
114. Tahir MA, Kittler J, Mikolajczyk K, and Yan F (2009) A multiple expert approach to the class imbalance problem using inverse random under sampling a multiple expert approach to the class imbalance problem using inverse random, no. May 2014
115. Moon TK (1996) The expectation-maximization algorithm. IEEE Sig Process Mag 13(6):47–60
116. Elzeki OM, Alrahmawy MF, and Elmougy S (2019) A new hybrid genetic and information gain algorithm for imputing missing values in cancer genes datasets, no. December, pp. 20–33
117. Azur MJ, Stuart EA, Frangakis C, Leaf PJ, and Washington DC (2011) Multiple imputation by chained equations: what is it and how does it work?, 20(1):40–49
118. Hotchkiss RS, Moldawer LL, Opal SM, Reinhart K, Turnbull IR, and Vincent JL (2016) Sepsis and septic shock. Nat Rev Dis Prim, 2
119. F. Mitchell M Levy, MD, MCCM Sean R Townsend, MD (2019) Early identification of sepsis on the hospital floors
120. Kaji DA et al (2019) An attention based deep learning model of clinical events in the intensive care unit. PLoS ONE 14(2):1–17
121. Snoek J, Larochelle H and Adams RP, Practical Bayesian optimization of machine learning algorithms, pp. 1–9
122. Bergstra J and Bengio Y (2012) Random search for hyper-parameter optimization, 13:281–305
123. Islam MM, Nasrin T, Walther BA, Wu CC, Yang HC, Li YC (2019) Prediction of sepsis patients using machine learning approach: a meta-analysis. Comput Methods Prog Biomed 170:1–9
124. Wang RZ, Sun CH, Schroeder PH, Ameko MK, Moore CC, Barnes LE (2018) Predictive models of sepsis in adult ICU patients. Proc - 2018 IEEE Int Conf Healthc Informatics, ICHI pp. 390–391
125. Deḿ J (2006) Statistical comparisons of classifiers over multiple data sets, 7:1–30

126. Friedman M (1990) The use of ranks to avoid the assumption of normality implicit in the analysis of variance. J Am Stat Assoc 32(200):675–701
127. Calvert J et al (2016) High-performance detection and early prediction of septic shock for alcohol-use disorder patients. Ann Med Surg 8:50–55
128. Shashikumar SP, Josef CS, Sharma A, Nemati S (2021) DeepAISE – an interpretable and recurrent neural survival model for early prediction of sepsis. Artif Intell Med 113:102036

Authors and Affiliations

Nora El-Rashidy[1] · Tamer Abuhmed[2] · Louai Alarabi[3] · Hazem M. El-Bakry[4] · Samir Abdelrazek[4] · Farman Ali[5] · Shaker El-Sappagh[6]

✉ Tamer Abuhmed
tamer@skku.edu

[1] Machine Learning and Information Retrieval Department, Faculty of Artificial Intelligence, Kafrelsheiksh University, Kafrelsheiksh 13518, Egypt

[2] College of Computing and Informatics, Sungkyunkwan University, Suwon, South Korea

[3] Computer Science Department, College of Computer Science and Information System, Umm Al-Qura University, Mecca, Saudi Arabia

[4] Information Systems Department, Faculty of Computers and Information, Mansoura University, Mansoura 13518, Egypt

[5] Department of Software, Sejong University, Seoul, South Korea

[6] Faculty of Computer Science and Engineering, Galala University, 435611 Suez, Egypt