Early Detecting In-Hospital Cardiac Arrest Based on Machine Learning on Imbalanced Data

Hsiao-Ko Chang
Department of Computer Science and Information Engineering
National Taiwan University
Taipei, Taiwan
d06922027@ntu.edu.tw

Cheng-Tse Wu
Department of Computer Science and Information Engineering
National Taiwan University
Taipei, Taiwan
d02922011@ntu.edu.tw

Ji-Han Liu
Graduate Institute of Networking and Multimedia
National Taiwan University
Taipei, Taiwan
d03944005@ntu.edu.tw

Wee Shin Lim
Department of Computer Science and Information Engineering
National Taiwan University
Taipei, Taiwan
leolim3092@csie.ntu.edu.tw

Hui-Chih Wang
Department of emergency medicine
National Taiwan University Hospital
Taipei, Taiwan
ticoer@ntuh.gov.tw

Shu-I Chiu
Department of Computer Science and Information Engineering
National Taiwan University
Taipei, Taiwan
sichiu@ntu.edu.tw

Jyh-Shing Roger Jang
Department of Computer Science and Information Engineering
National Taiwan University
Taipei, Taiwan
jang@csie.ntu.edu.tw

***Abstract*—In-hospital cardiac arrest (IHCA) diminish the survival rate of patients, despite most of the IHCA cases are preventable. More than 54% IHCA patient had abnormal clinical manifestation before they suffered a cardiac arrest. If appropriate steps were taken, patients' survival rate would be higher and medical expense would be decreased. This paper proposes a novel approach to detect IHCA before the event occurred. We construct two types of shifting windows (corresponding to two tasks) that allow machine learning to be applied for our dataset which is severely imbalanced. The results show that our approach can effectively handle the imbalanced dataset for detecting cardiac arrest. As the selection of performance index, we used the area under the receiver operating characteristic curve (AUROC) and the area under the precision–recall curve (AUPRC). In our experiments, the best classifier is random forest for task 1, with AUROC of 0.88. LSTM is the best for task 2, with AUPRC of 0.71 for the second task.**

Keywords—Cardiac arrest, cardiopulmonary resuscitation, imbalanced data classification, machine learning, prediction

I. INTRODUCTION

Cardiac arrest is a sudden loss of blood flow resulting from the failure of the heart to effectively pump. In-hospital cardiac arrest (IHCA) is a major burden to public health, which affects patient safety [1]. 80% of patients with cardiac arrest show signs of deterioration in the 8 hours before cardiac arrest [1]. Cardiopulmonary resuscitation (CPR) is originally developed for victims of sudden cardiac or respiratory arrest [2]. Doing CPR keeps blood circulating until trained to jump-start the heart back into a normal rhythm. This technique came into being after the invention of closed-chest cardiac massage in 1960 [2, 3]. It is an emergency procedure that combines chest compressions often with artificial ventilation in an effort to manually preserve intact brain function until further measures are taken to restore spontaneous blood circulation and breathing in a person who is in cardiac arrest [4]. It is standard practice to attempt CPR on any patient in the hospital who has a cardiac arrest, regardless of the underlying illness [2]. In the United States, 209,000 IHCA occur each year, and the survival discharge rate for patients with cardiac arrest is <20% worldwide [1]. Therefore, it is very important to detect IHCA. We design a novel approach to detect IHCA for emergency patients. Our goal is to develop an early warning system (EWS) to avoid using CPR for patients. We propose some methods to develop the EWS.

Cardiac arrest and major trauma are relatively common in emergency departments (EDs). CPR technique is performed heavily in hospitals. In this paper, we focus on emergency patients. While patients can present at any time and with any complaint, a key part of the operation of an ED is the prioritization of cases based on clinical need. This process is called triage [5]. Triage is normally the first stage the patient passes through, and consists of a brief assessment, including a set of vital signs. In Taiwan, the five-level Taiwan Triage and Acuity Scale (TTAS) computerized system was implemented nationally in 2010. The TTAS retains most features of the Canadian triage and acuity scale (CTAS) [6]. Our dataset contains this value with a computerized decision support system. In recent years, a great amount of electronic health records (EHRs) have become available [7]. EHRs are collections both static and dynamic features. The static features contain patient background data. On the other hand, the dynamic features, such as lab tests and vital signs, are collected multiple time during a patient's visit [7, 8]. Generally speaking, the dynamic features are represented as a time series. Our dataset also contains static and dynamic features during a patient's visit. The static features include age, gender, weight, and triage (i.e., TTAS); the dynamic ones include vital signs and drug information. We aim to detect IHCA by combining static and dynamic features to test models.

The classical data imbalance problem is recognized as one of the major problems in the field data mining and machine learning as most machine learning algorithms assume that data is equally distributed. In the case of imbalanced data, majority classes dominate over minority classes, causing the machine learning classifiers to be more biased towards majority classes. This causes poor classification of minority classes. Classifiers may even

predict all the test data as majority classes. Some of the real-world examples involve fraud detection in banking, intrusion detection in networks, and rare diseases [9, 10, 11]. However, imbalanced class distribution of a dataset has encountered a serious difficulty to most classifier learning algorithms which assume a relatively balanced distribution [12]. Most machine learning algorithms do not work well with imbalanced datasets. We propose a novel approach to deal with such datasets. In this paper, we design shifting windows that are time-based. We define a shifting window as collected EHRs during a fixed time for each patient.

As adoption of artificial intelligence and machine learning becomes more pervasive, the way we live and work is being fundamentally altered. Neural network is one of the most popular machine learning algorithms today. It has been decisively proven that neural networks outperform other algorithms in accuracy and speed. Recurrent Neural Networks (RNNs) have proven to be very successful for modelling sequences of data in many areas of machine learning [8]. Long Short-Term Memory (LSTM) networks are an extension for RNNs, which basically extends their memory. Therefore it is well suited to learn from important experiences that have very long time lags in between. LSTM is effective in capturing underlying temporal structures in time series data [13]. It makes the model particularly suitable at modeling dynamic information in EHRs, where there is a strong statistical dependency between medical events over long-time intervals [7]. Our proposed method applies classic machine learning algorithms and LSTM to imbalanced data learning.

Three major contributions of our work are:

- To handle imbalanced ratio, we design shifting windows to adjust our dataset and propose a novel approach exploring disease detection.
- We combine static and dynamic features to detect IHCA. We construct sequences of dynamic data for neural networks. Our goal is to develop some models to apply to an EWS.
- This study explores the combined efficacy of these two components: Convolutional Neural Network (CNN) and LSTM. CNN is added before LSTM to obtain static features; LSTM is applied to handle dynamic features. Our proposed approach uses not only classic machine learning classifiers but also neural networks to detect IHCA.

The remainder of this paper is organized as follows: We will briefly review the related works in Section 2, describe the proposed methods in Section 3, depict experiments in Section 4, report results from experiments in Section 5, and finally conclude this paper in Section 6.

II. Related Work

Annual adult non-traumatic patients who stay in emergency room more than 6 hours in a tertiary medical center were enroll in this research. Patients who signed do not resuscitation (DNR) were excluded and those who alive on arrival but received CPR during stay in emergency room were IHCA patients. Initially designed to rescue patients experiencing a sudden cardiac arrest due to arrhythmia, CPR has come to be seen as a procedure that should be used for patients for whom there is a reasonable chance of restoring cardiopulmonary function and prolonging life [14]. There are studies, such as [1, 2, 3, 4, 14, 15], where CPR technique is an important procedure to improve survival for sudden cardiac arrest. Its main purpose is to restore partial flow of oxygenated blood to the brain and heart. The objective is to delay tissue death and to extend the brief window of opportunity for a successful resuscitation without permanent brain damage [16].

With the approach of big data epoch, people can benefit from large-scale and real-time data to help improve the diagnosis decision, applying data mining and machine learning techniques. However, in many real-life problems, especially in the medical field, the datasets are commonly imbalanced. The performance of classifier would be terribly affected when imbalanced data is not managed well [17]. If you take the example of rare diseases, machine learning may suffer from accuracy paradox, which makes it difficult to control false positives and false negatives. On the other hand, patients may suffer from a rare disease but the machine learning models do not predict. The results become most patients without disease in the dataset.

To overcome this problem, some approaches have been proposed that can be implemented during the pre-processing stage. One commonly used strategy is called resampling, which includes under-sampling and over-sampling. Over-sampling can be achieved by adding similar data of underrepresented class to balance the skewed class ratio. There are a number of methods to over-sample a dataset used in the typical classification problem. The common techniques are known as Synthetic Minority Over-sampling Technique (SMOTE) [18], Borderline-SMOTE [19], and Adaptive Synthetic Sampling Approach (ADASYN) [20]. There are some methods to under-sample data. The common techniques are Cluster, random sampling, and stratified sampling. In this paper, we address a novel approach to add similar data of underrepresented class to balance the class ratio. Our approach differs from SMOTE and ADASYN in expanding data coming from the original data. By the shifting window construction, we expand the amount of IHCA.

For the imbalanced problem, some researches focus on diagnoses or diseases. In [17], they propose an ensemble model to precisely diagnose the diabetic on a large-scale and imbalance dataset. They make efforts to reduce the variance by under-sampling as much as possible [17]. It is well known that an excellent model requires both low variance and low bias [17]. They build a classifier to fit the resampled data aiming at decreasing the bias. The information required for medicine diagnosis is typically collected from a history and physical examination of the person seeking medical care. Cardiac arrest is different from diabetes disease and it is a sudden loss of blood flow resulting from the failure of the heart to effectively pump.

On the other hand, some studies [21, 22, 23] handle multi-class imbalanced problem. In [22], they design two-stage adaptive weighted extreme learning machine method. Their approach achieves a good balance between high detection accuracy and low false-alarm rate based on our two-stage recognition scheme [22]. In [23], they design a complete, fully automatic and efficient clinical decision support system for breast cancer malignancy grading. In order to overcome the imbalanced classification problem, they propose the usage of an efficient ensemble classifier

named EUSBoost, which combines a boosting scheme with evolutionary under-sampling for producing balanced training sets for each one of the base classifiers in the final ensemble [23].

For a binary class problem, the imbalance degree of a class distribution can be denoted by the ratio of the sample size of the small class to that of the prevalent class [12]. In practical applications, the ratio can be as drastic as 1:100, 1:1000, or even larger [24]. In [25], research is conducted to explore the relationship between the class distribution of a training dataset and the classification performances of decision trees. Their study indicates that a relatively balanced distribution usually attains a better result [12]. In some applications, a ratio as low as 1 : 35 can make some methods inadequate for building a good model [12]. The ratio of our original dataset (IHCA positive patients : IHCA negative ones) is 1 : 350. IHCA positive patients denote patients who need to use CPR in the ED. We under-sample IHCA negative patients for detection; the ratio of adjusted dataset becomes 1 : 10. The IHCA negative patients denote patients who do not use CPR during patients' visit in the ED.

The neural network itself is not an algorithms, but rather a framework for many different machine learning algorithms to work together and process complex data inputs. RNN is one of the most extensively researched deep neural networks for handling temporal sequential data [7]. LSTM is a type of RNN specifically designed to avoid gradient vanishing and exploding problems. RNN has been applied to many EHR applications due to its memory maintenance mechanism [26] and parameter sharing scheme, which allow the model to capture long-range temporal dependency and to handle sequences of varying length. We apply LSTM to test our models.

III. METHODS

Data Descriptions. This study uses EHRs of ED obtained from a public teaching hospital which is one of the greatest hospitals in Taiwan. This dataset is collected EHRs for adult patients (age ≧20 years) visited ED from 2014 to 2015. It covers 2 whole years. We collect non-traumatic patients excluding do not resuscitate; these patients stay ED more than 6 hours. Each patient information was anonymized and deidentified before the analysis.

Our dataset contains both static and dynamic features. The list of static and dynamic features is shown in Table I. Static features include patient background information collected once per visit; our study contains 9 static ones. Dynamic features are collected multiple times at irregular intervals during the patients' registered hospitalization and have a time stamp associated with each record. Hence, dynamic features are expressed as a temporal sequence; this study contains 10 dynamic ones.

TABLE I. STATIC AND DYNAMIC FEATURES

Types	Features
Static	Age, gender, height, weight, fever, Glasgow Coma Scale (eye opening, verbal response, and motor response), and triage (i.e., TTAS).
Dynamic	➢ Vital signs: mean arterial pressure (MAP), systolic blood pressure, diastolic blood pressure, pulse, respiratory rate, and body temperature (BT). ➢ Drug information: Intravenous therapy (IV) injection, painkiller, antibiotic, and diuretic.

Study Population The study population are emergency patients who stay ED more than 6 hours. These patients contain static and dynamic features during their visits. Because human errors could exist in the EHRs, we exclude systolic blood pressure, diastolic blood pressure, pulse, respiratory rate, and BT values that were outside the ranges of 40 to 300 mm Hg, 20 to 300 mm Hg, 30 to 200 beats/min, 3 to 60 breaths/min, and 28 to 42°C, respectively. We identify 107 IHCA positive patients and 28,953 IHCA negative patients in the ED. We conduct a stratified random sampling on IHCA negative patients while keeping the same underlying distribution of age, gender, and length of stay. We under-sample IHCA negative patients and the ratio of adjusted dataset becomes 1 : 10. As a result, the adjusted dataset contains 1,177 patients (107 positives and 1,070 negatives).

The workflow for the processing is illustrated in Fig. 1. Due to the imbalanced ratio, IHCA negative patients are under-sampled by stratified random sampling. The sampled data and original data are the same distribution.

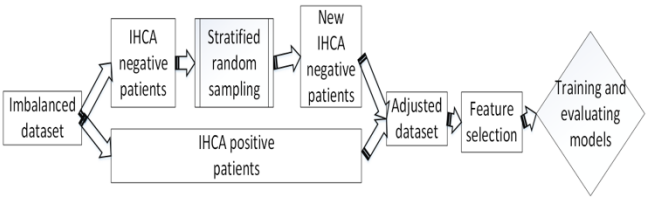

Fig. 1. The data flow

In this paper, we design the shifting window for dynamic features. We define a shifting window as collected EHRs during a fixed time (i.e., *m* hours) for each patient's visit in Fig. 2. Fig. 2 shows a IHCA positive patient and a shifting window including EHRs. These EHRs contain vital signs and drug information. In Fig. 2, the CPR time is the first time using CPR technique during a patient's visit. On the other hand, a shifting window is composed of a sequence of partition. Each partition collects a number of EHRs. In this paper, we also define the length of the shifting window and the shifting time between any two adjacent shifting windows. In Fig. 2, *m* hours is the length of a shifting window. When the shifting window is advanced, the oldest partition is disregarded and a new partition containing a set of newly collected EHRs are appended to the window. Shifting windows are shifted by a fixed time. This is the shifting time. In our experiments, we adjust various lengths of a shifting window. On the other hand, we use EHRs in a shifting window to detect whether a patient uses CPR technique 1 hour later or not. Shifting windows can apply to LSTM based on time series data.

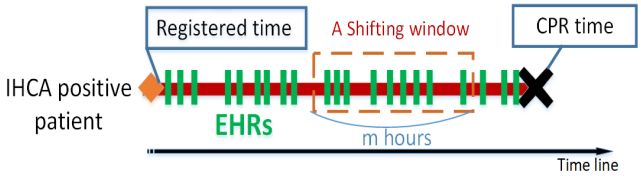

Fig. 2. A shifting window

The dynamic features are represented as a time series. A shifting window may contain at least one EHR for any feature. In order to display dynamic features information completely, we calculate dynamic features statistics. During the specific shifting window, the mean, the first quartile (Q1), the third quartile (Q3), maximum, minimum, and standard deviation of every dynamic feature are calculated as new variables. In addition, the first and last records of each dynamic feature are captured as new variables. Consequently, each dynamic feature is generated 8 new features in a shifting window.

Feature Selection In machine learning and statistics, feature selection is the process of selecting a subset of relevant features for use in model construction. Each patient contains many shifting windows. A shifting window includes 8 new variables for each dynamic features in order to display dynamic feature information completely. There are many static and dynamic features including generated new features for each shifting window. Therefore, we use the sequential forward selection (SFS) manner by Whitney [27] for feature selection. SFS is one of the commonly used heuristic methods for feature selection. We use *k*-Nearest Neighbor (*k*NN) [28] and the leave-one-out test for F_3 score estimate. After adjusted dataset is used the feature selection method to reduce features.

$$f(xi) = \frac{xi - \min(\forall xi)}{\max(\forall xi) - \min(\forall xi)} \quad (1)$$

However, the ranges of the values of the dynamic features are large. This causes difficulty for classifier training. Therefore, we use Normalization function to adjust the values of the static and dynamic features in Equation 1. Due to various ranges of the values of these features, we only use the normalization function to adjust these values. The range of these normalized values is between 0 and 1.

IV. EXPERIMENTS

The machine learning is implemented using the scikit-learn package in Python [30]; the neural networks are implemented in Keras[1] with TensorFlow[2] as the backend engine. The scikit-learn package also implements multiple classification problems. For machine learning, regarding classification algorithms, we use top ones [31]: Naïve Bayes [32, 33], Support Vector Machines (SVM) [34 , 35], AdaBoost [36, 37] and C4.5 Decision Tree [38]. In addition, we also use the Random Forest algorithm [39]. Random forests are an ensemble learning for classification, regression, and other tasks by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes or mean prediction of the individual trees. For a LSTM layer, we experiment 20, 40, 60, and 80 units and the best one was chosen. Categorical Cross-entropy is applied as loss function and Adam optimizer is used for optimization. It is used for multi-class classification. All models are evaluated using leave-one-out cross-validation. We compare the classification performance given by the two tasks described previously and these classification algorithms.

All models are tested using two tasks. The two tasks focus on time series data.

[1] https://keras.io/

[2] https://www.tensorflow.org/

A. *The shifting window without overlap (the first task)*

We design shifting windows; this task is to predict whether the patient will use CPR *m* hours later. To follow out this task, the patients' sequences are right aligned. For IHCA positive patients, the end point is the onset time of ones using IHCA. For IHCA negative patients, the end point is not the end of sequences. To detect the two classes, we calculate the mean of difference between the time when a IHCA patient registers with the hospital and the time when this patient uses CPR for the first time. The two classes have similar observation time. Then, we construct some *m*-hour shifting windows for every patient. Fig. 3 illustrates this task. We need for classifier training. We label each emergency patient as *P* (for IHCA positives) or *N* (for IHCA negatives). Therefore, after data processing, the task becomes a binary classification task.

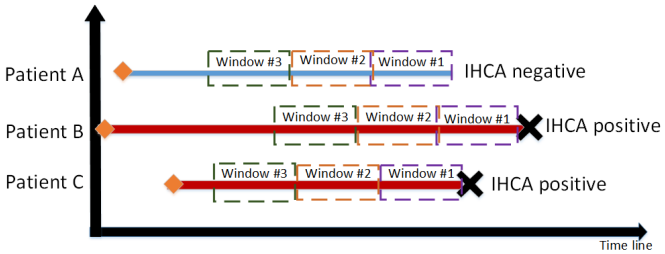

Fig. 3. The shifting window without overlap

When an emergency patient visit the hospital, medical staffs need to measure the patient's information, such as vital signs. We analyze information of every patient during m hours window to detect it. Fig. 3 illustrates this task. The rectangle with the dotted line is denoted as a shifting window; there are 3 shifting windows in Fig. 3. For IHCA positive patients, the window #1 is a shifting window of the nearest CPR time. In our experiments, the shifting window is denoted as 8-hour window, because medical staffs of hospitals work in shifts. The vital signs of the general patients were measured at least 3 times per day manually by the medical staff. In addition, most drugs are taken one dose four times a day. The time interval is about 6 hours. The few drugs are taken one dose six times a day; the time interval is 4 hours. Consequently, in the two tasks, the experiments are designed to detect IHCA using 4-, 6-, and 8-hour shifting window.

B. *The shifting window without overlap (the second task)*

In this task, we build shifting windows with overlap in Fig. 4. The shifting time between any two adjacent shifting windows is 1 hour. For IHCA positive patients, they will use CPR 1 hour later during a shifting window, They are labeled *P*. However, during a shifting window, a IHCA positive patient will not use CPR 1 hour later. They are labeled *U*. This means that the patient will use CPR at least 2 hours later during the shifting window. He/she is a potential IHCA positive patient. For IHCA negative patients, they are labeled *N* during any shifting window. In Fig. 4, during the shifting window #n, Patient A is labeled *N*; Patient B is labeled *P*. However, Patient C is a IHCA positive patient but Patient C will not use CPR 1 hour later during a shifting window(i.e., window #n). Patient C is labeled *U*. This class (i.e., patients are labeled *U*) collects records of many positive patients. This task becomes a multiple classification task after data processing. However, we transfer our dataset into binary

classification in order to train neural networks because our goal is to detect IHCA.

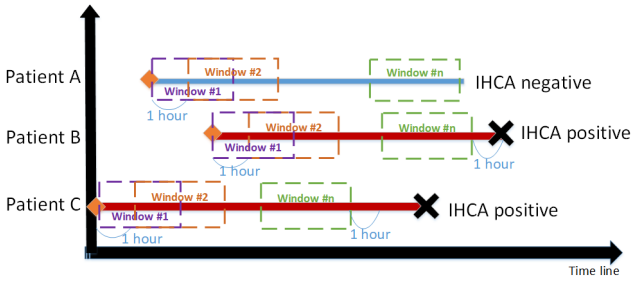

Fig. 4. The shifting window with overlap

C. Evaluation Metrics

We use the area under the receiver operating characteristic curve (AUROC), the area under the precision-recall curve (AUPRC), and F-Score (see Equation 2). F-Score is the harmonic mean of precision and recall and gives a good combination of the two [28]. Generally speaking, F-Score with β=3 is to emphasize recall.

$$F_\beta \text{ Score} = \frac{(\beta^2 + 1) \times precision \times recall}{\beta^2 \times precision + recall} \quad (2)$$

Furthermore, we use 10-fold cross-validation. It divides the dataset into 10 disjoint subsets. It uses 9 subsets to create a new dataset, and use the new dataset to train a classifier. Then, it uses the remaining 1 subset to test the classifier. It repeats the above two steps 10 times, and each time it uses a different subset. The final result is an aggregate of the 10 test results. Cross-validation is almost the standard way to evaluate classifiers and compare classification algorithms (and find an optimal set of parameters for a classification algorithm) in data mining.

V. Results

In this section, we present the results for both experimental settings.

A. The shifting window without overlap

In this task, how many shifting windows do we construct for a patient? For a patient, we compute the difference between the time when a patient registers with the hospital and the time when a patient use CPR for the first time. The mean of all patients' difference in time is 30 hours. On average, while a IHCA positive patient registers with the hospital, he/she will use CPR 30 hours later. Consequently, we depend on the average to adjust the amount of shifting windows. For instance, for 4-hour shifting window, we construct 7 shifting windows for each patient. In this paper, our goal is to detect IHCA early.

We depend on various lengths of shifting window to generated data. Table II shows results of feature selection. These selected features are sorted in Table II. Totally, pulse is the most important feature for 4- and 6-hour shifting window; systolic blood pressure is the important one for 8-hour shifting window. Pulse, respiratory rate, diuretic, BT, IV injection, painkiller are the more important features for 4-, 6-, and 8-hour shifting window. In this task, we rely on these features to detect IHCA.

TABLE II. Results of Feature Selection

Shifting window	Features
4-hour	Pulse, respiratory rate, diuretic, BT, IV injection, painkiller, gender, Glasgow Coma Scale.
6-hour	Pulse, respiratory rate, diuretic, IV injection, Glasgow Coma Scale, gender, systolic blood pressure, painkiller, MAP, antibiotic, BT, TTAS.
8-hour	Systolic blood pressure, pulse, respiratory rate, BT, IV injection, diuretic, painkiller.

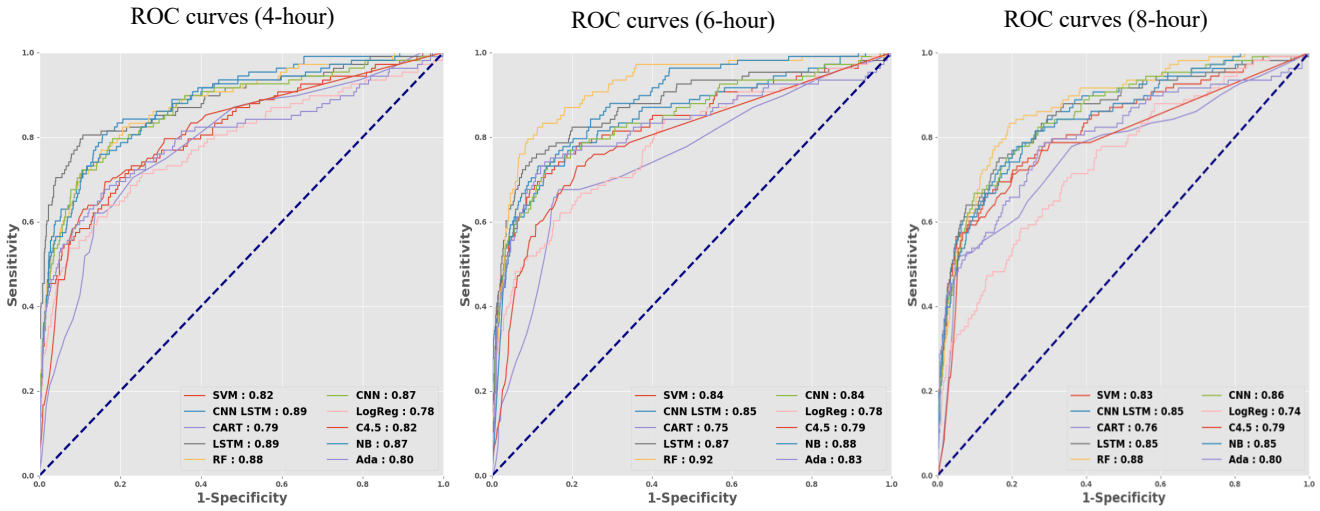

Fig. 5. The various lengths of the shifting window without overlap (ROC curves)

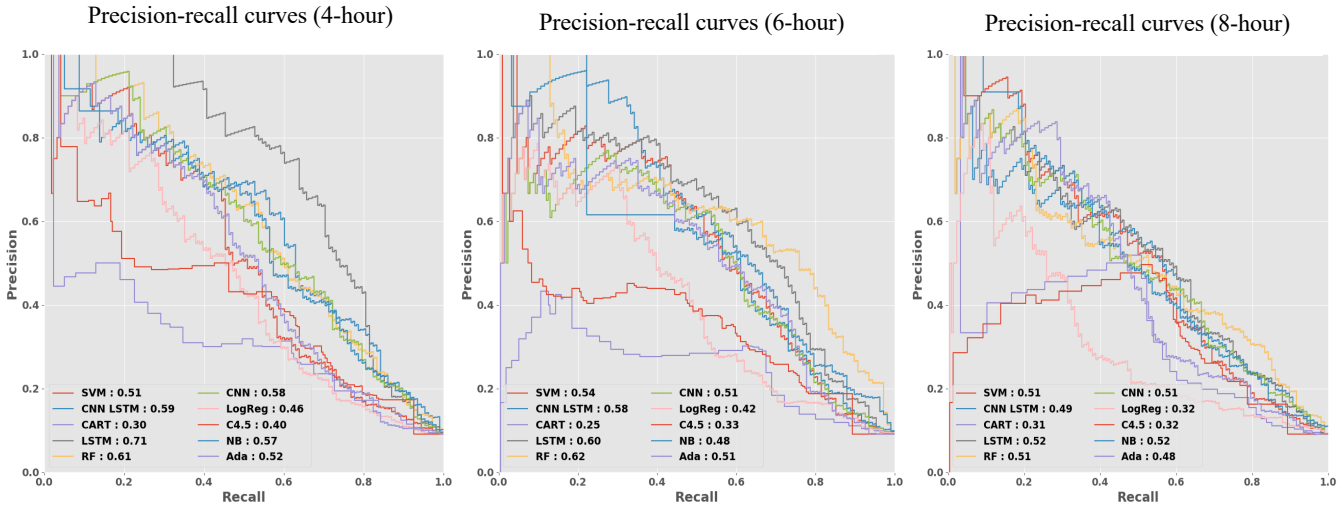

Fig. 6. The various lengths of the shifting window without overlap (precision-recall curves)

Fig. 5 and 6 show ROC curves and precision-recall curves for the shifting window without overlap while varying lengths of shifting window are from 4 hours to 8 hours.

The higher the AUROC value is, the better the model is. For 4-hour shifting window, the best classifier is the random forest; the AUROC value is 0.88. CNN+LSTM is best and the AUROC value is closed to 0.9. For 6-hour shifting window, the best classifiers are random forest and naïve Bayes but CNN+LSTM would not perform well. The best classifier is random forest for 8-hour shifting window. Totally, the performance for 4-hour shifting window is better than the performance for 6- and 8-hour shifting window. For the 4-hour shifting window, window #1 is a shifting window of the nearest CPR time in Fig. 3. Its performance is the best. Regardless of the length of shifting window, the best classifiers are random forest and CNN+LSTM.

In general, precision-recall curves are often zigzag curves frequently going up and down. Precision-recall curves tend to cross each other much more frequently than ROC curves. Similarly, the higher the AUPRC value is, the better the model is. For 4-hour shifting window, LSTM is the best and the AUPRC value is 0.71. This value is the highest regardless of the length of shifting window. Random forest is the best for 6-hour shifting window; LSTM and Naïve Bayes are the best for 8-hour shifting window.

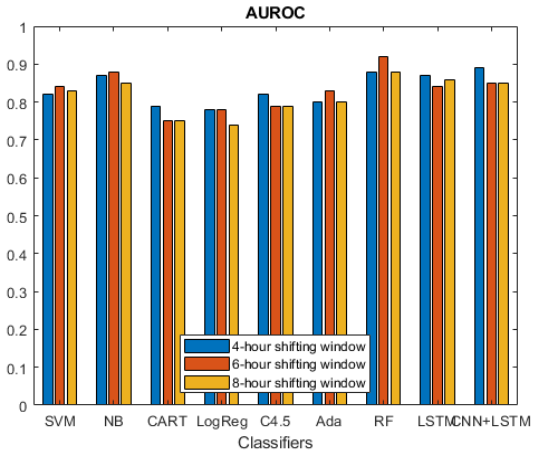

Fig. 7. AUROC values given by machine learning classifiers and neural networks

We compare the 4-, 6-, and 8-hour shifting window. The results in Fig. 7. Totally, for machine learning classifiers, 6-hour shifting window performs well; for neural networks, 4-hour shifting window performs well. On the other hand, the most amount of shifting windows performs well for neural networks.

In addition, we calculate F_3 scores that are a good combination of precision and recall. The results are given in Fig. 8. F_3 score has emphasizes recall over precision, with 3 indicating that recall is weighted three times as much as precision. In this task, random forest is best during 6-hour shifting window; this F_3 score is 0.75. Totally, the 6-hour shifting window performs well. Regardless of the length of shifting window, the best classifiers are random forest and naïve Bayes.

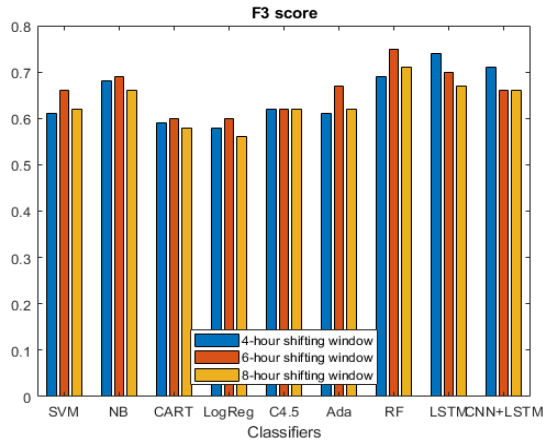

Fig. 8. F_3 scores given by machine learning classifiers and neural networks

B. The shifting window with overlap

The adjusted dataset has 1,177 patients: 107 IHCA positive and 1,070 IHCA negative ones. In this task, the shifting time is set 1 hour. For classic machine learning algorithms, this tasks belong to multiple class problem in Fig. 9, 10, 11, and 12. We define 3 classes in Fig 4. In our experiments, we also choose random forest classifier to analyze the 4-, 6-, and 8-hour shifting window.

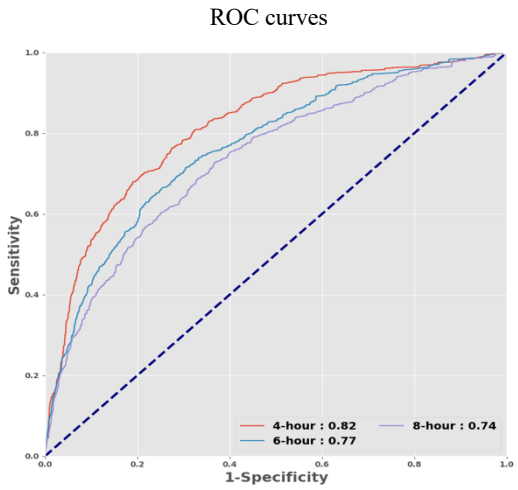

Fig. 9. The various lengths of shifting window with overlap (ROC curves)

On AUROC and AUPRC performance, the 4-hour shifting window is the best and the 6-hour shifting window is the second in Fig. 9 and 10. The AUPRC value is 0.82 and the AUPRC value is 0.78 for the 4-hour shifting window. Because the 4-hour shifting window is the nearest CPR time, it perform well.

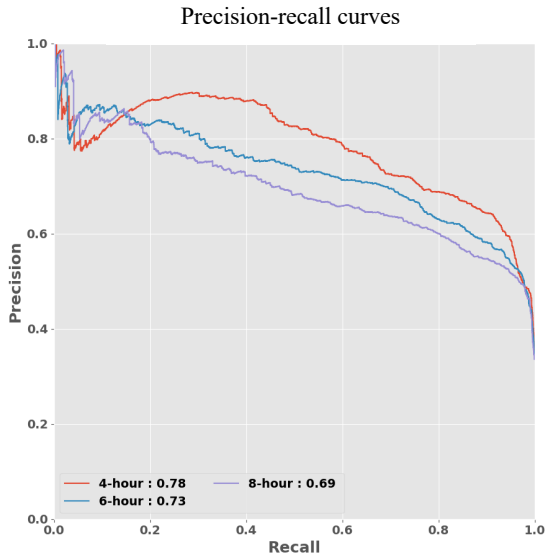

Fig. 10. The various lengths of shifting window with overlap (precision-recall curves)

We rely on the 4-hour shifting window with overlap to test all models including machine learning algorithms and neural networks. Fig. 11 and 12 present the ROC and precision and recall curves, respectively.

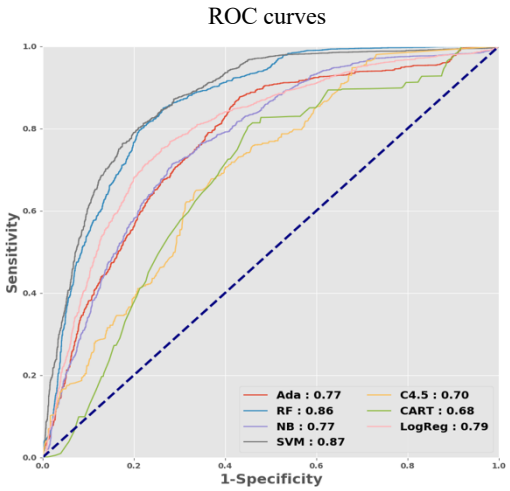

Fig. 11. The different classifiers of the 4-hour shifting window with overlap (ROC curves)

Fig. 11 shows that the best classifiers are SVM and random forest. The AUROC values are 0.87 and 0.86, respectively. Fig. 12 shows that the best classifiers are also SVM and random forest. The AUPRC values are 0.73 and 0.7, respectively. In the previous studies [1, 7], random forest was the most commonly used machine learning algorithms and showed better performance than others.

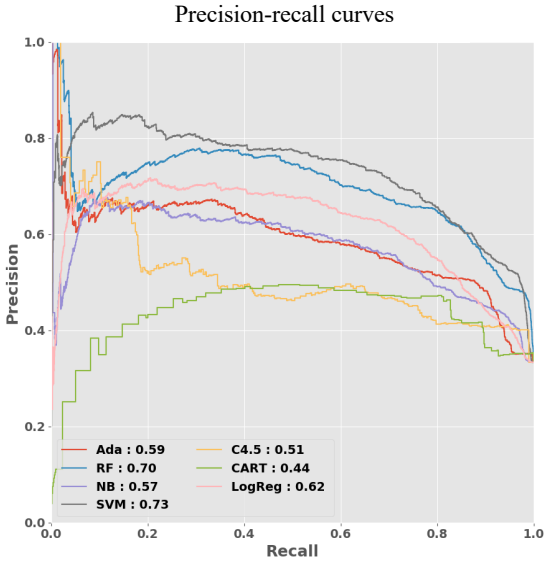

Fig. 12. The different classifiers of the 4-hour shifting window with overlap (precision-recall curves)

Fig. 13 and 14 show the ROC and precision-recall curves for neural networks. It belongs to binary classification problem. Because neural networks need time series data, we adjust the 3-class (i.e., labeled *P*, *U*, and *N*) dataset to the 2-class (i.e., labeled *P* and *N*) dataset. CNN performs the best and the AUROC value is 0.77 in Fig. 13. LSTM is the best and the AUPRC value is 0.3 in Fig. 14.

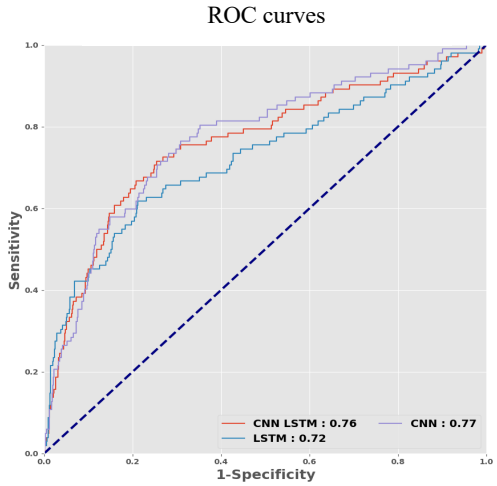

Fig. 13. The different nerual networkss of the 4-hour shifting window with overlap (ROC curves)

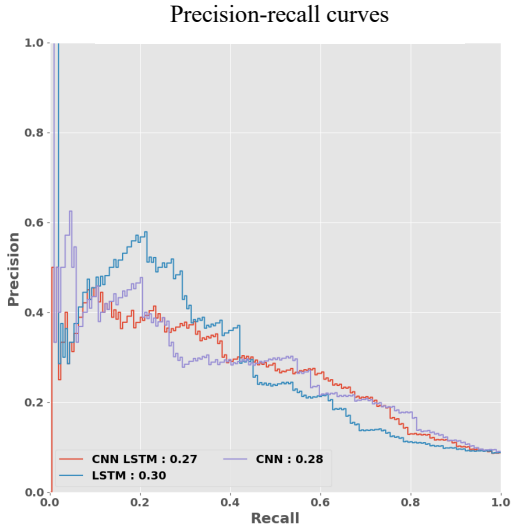

Fig. 14. The different nerual networkss of the 4-hour shifting window with overlap (precision-recall curves)

C. *Comparison*

To compare the two tasks, we use two performance measures, namely, AUROC and AUPRC. In Fig. 15, we present the best settings (or the settings giving the best results) chosen from Fig. 5, 6, 9, and 10, and we also present the best classifier for each of the two tasks. For the first task (i.e., the shifting window without overlap), we choose the 4-hour shifting window; for the second task (i.e., the shifting window with overlap), we choose the 4-hour shifting window.

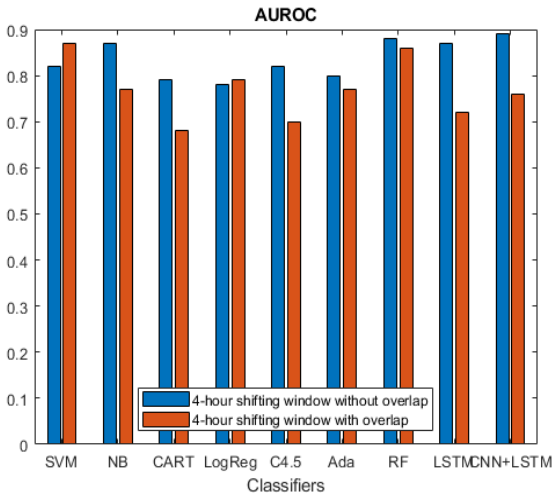

Fig. 15. Comparison in AUROC performance measure between the two tasks

According to Fig. 15, the first task is better than the second task in the AUROC performance measure totally. For 4-hour shifting window without overlap, the best classifier is Naïve Bayes and random forest. For 4-hour shifting window with overlap, the best classifier is random forest. Consequently, random forest is the best classifier for detection of IHCA. In the previous studies, logistic regression and random forest were the most commonly used machine learning methods and showed better performance than traditional algorithms [1].

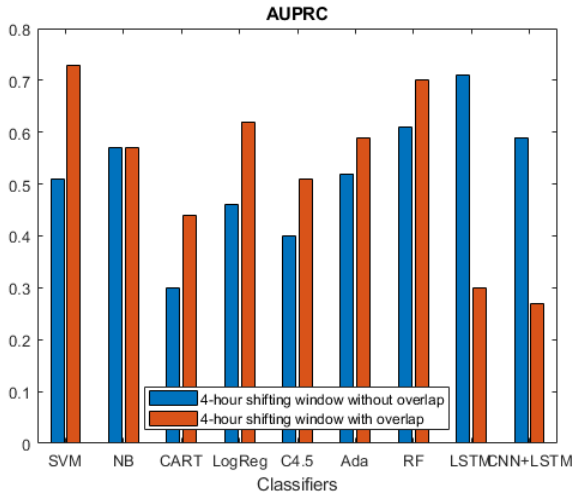

Fig. 16. Comparison in AUPRC performance measure between the two tasks

According to Fig. 16, the second task is better than the first task in the AUPRC performance measure totally. For 4-hour shifting window without overlap, the best classifier is random forest. For 4-hour shifting window with overlap, the best classifier is also random forest. Consequently, random forest is the best classifier for detection IHCA.

We compare the two tasks and present the comparison in Table III. The first task represents deep neural networks. The second task can be applied to other disease detection. The first task represents a better method for AUROC; the second task represents a better method for AUPRC.

TABLE III. COMPARISON BETWEEN THE TWO TASKS

	The first task	The second task
Advantage	Can be applied to neural networks	General and can be applied to other disease detection and can handle imbalanced data
Disadvantage	Runs slower and needs more memory	Cannot be applied to neural networks
Application	Can detect patients whether are IHCA or not	Can develop an EWS
Classification	Two classes	Three classes for classic machine learning algorithms; two classes for neural networks
AUROC	Higher	Lower
AUPRC	Lower	Higher

VI. CONCLUSION

Machine learning is one of the most exciting technologies that one would have ever come across. It can be explained as automating and improving the learning process of computers based on their experiences without any human assistance. We apply classifiers to detect IHCA by combining static and dynamic features. We use two tasks to test all models. For the shifting window without overlap, 4-hour shifting window is the best performance. The best classifier is random forest totally and the AUROC value is 0.88. CNN+LSTM is best; the AUROC value is closed to 0.9. For the shifting window with overlap, 4-hour shifting window is also the best performance. LSTM is the best and the AUPRC value is 0.71. In the future, we will add more features to our models.

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to the anonymous reviewers for their constructive comments on the manuscript of this paper. The work presented in this paper was supported in part by the Ministry of Science and Technology, R.O.C., under grant number 107-2634-F-002-015. The authors acknowledge the support.

REFERENCES

[1] J. M. Kwon, Y. Lee, Y. Lee, S. Lee, and J. Park, "An Algorithm Based on Deep Learning for Predicting In‐Hospital Cardiac Arrest," *Journal of the American Heart Association*, vol. 7, no. 13, e008678, 2018.

[2] L. J. Blackhall, "Must we always use CPR," *New England Journal Medicine*, vol. 317, no. 20, pp. 1281-1285, 1987.

[3] W. B. Kouwenhoven, J. R. Jude, and G. G. Knickerbocker, "Closed-chest cardiac massage," *JAMA*, vol. 173, no. 10, pp. 1064-1067, 1960.

[4] D. L. Atkins, et al., "Part 11: pediatric basic life support and cardiopulmonary resuscitation quality: 2015 American Heart Association guidelines update for cardiopulmonary resuscitation and emergency cardiovascular care," *Circulation*, 132(18 suppl 2), pp. S519-S525.

[5] D. Kindermann, R. Mutter, and J. M. Pines, "Emergency Department Transfers to Acute Care Facilities," *HCUP Statistical Brief* #157, 2013.

[6] C. J. Ng, Z. S. Yen, J. C. H. Tsai, L. C. Chen, S. J. Lin, Y. Y. Sang, and J. C. Chen, "Validation of the Taiwan triage and acuity scale: a new computerised five-level triage system," *Emerg Med J*, vol. 28, no. 12, pp. 1026-1031, 2011.

[7] C. Lin, Y. Zhangy, J. Ivy, M. Capan, R. Arnold, J. M. Huddleston, and M. Chi, "Early Diagnosis and Prediction of Sepsis Shock by Combining Static and Dynamic Information Using Convolutional-LSTM," in *Healthcare Informatics (ICHI), 2018 IEEE International Conference on.* IEEE, 2018, pp. 219-228.

[8] C. Esteban, O. Staeck, S. Baier, Y. Yang, and V. Tresp, "Predicting clinical events by combining static and dynamic information using recurrent neural networks," in *Healthcare Informatics (ICHI), 2016 IEEE International Conference on.* IEEE, 2016, pp. 93-101.

[9] X. Guo, Y. Yin, C. Dong, G. Yang, and G. Zhou, "On the class imbalance problem," in *Natural Computation, 2008. ICNC'08. Fourth International Conference on (Vol. 4, pp. 192-201)*. IEEE.

[10] S. Kotsiantis, D. Kanellopoulos, and P. Pintelas, "Handling imbalanced datasets: A review," *GESTS International Transactions on Computer Science and Engineering*, vol. 30, no. 1, pp. 25-36, 2006.

[11] S. Visa and A. Ralescu, "Issues in mining imbalanced data sets-a review paper," in *Proceedings of the sixteen midwest artificial intelligence and cognitive science conference* (Vol. 2005, pp. 67-73). sn.

[12] Y. Sun, A. K. Wong, and M. S. Kamel, "Classification of imbalanced data: A review," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 23, no.04, pp. 687-719, 2009.

[13] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735-1780, 1997.

[14] S. J. Goodlin, et al., "Factors Associated with Use of Cardiopulmonary Resuscitation in Seriously Ill Hospitalized Adults," *JAMA*, vol. 282, no. 24, pp. 2333-2339, 1999.

[15] B. Zakhary, V. B. Nanjayya, J. Sheldrake, K. Collins, J. F. Ihle, and V. Pellegrino, "Predictors of mortality after extracorporeal cardiopulmonary resuscitation," *Critical Care and Resuscitation*, vol. 20, no. 3, pp. 223-230, 2018.

[16] J. J. Mistovich, K. J. Karren, and B. Hafen, Prehospital Emergency Care, 10e, Pearson Education, Inc., 2014.

[17] X. Wei, F. Jiang, F. Wei, J. Zhang, W. Liao, and S. Cheng, "An ensemble model for diabetes diagnosis in large-scale and imbalanced dataset," in *Proceedings of the computing frontiers conference*, 2017, pp. 71-78.

[18] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol 16, pp. 321-357, 2002.

[19] H. Han, W. Y. Wang, and B. H. Mao, "Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning," in *International Conference on Intelligent Computing*, 2005, pp. 878-887.

[20] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in *Neural Networks, 2008. IJCNN 2008*, pp. 1322-1328.

[21] B. Krawczyk, "Learning from imbalanced data: open challenges and future directions," *Progress in Artificial Intelligence*, vol. 5, no. 4, pp. 221-232, 2016.

[22] X. Gao, Z. Chen, S. Tang, Y. Zhang, and J. Li, "Adaptive weighted imbalance learning with application to abnormal activity recognition," *Neurocomputing*, vol. 173, pp. 1927-1935, 2016.

[23] B. Krawczyk, M. Galar, Ł. Jeleń, and F. Herrera, "Evolutionary undersampling boosting for imbalanced classification of breast cancer malignancy," *Applied Soft Computing*, vol. 38, pp. 714-726, 2016.

[24] N. V. Chawla, N. Japkowicz and A. Kolcz, "Editorial: special issue on learning from imbalanced data sets," *SIGKDD Explorations Special Issue on Learning from Imbalanced Datasets*, vol. 6, no. 1, pp. 1-6, 2004.

[25] G. Weiss and F. Provost, "Learning when training data are costly: the effect of class distribution on tree induction," *Journal of Artificial Intelligence Research*, vol. 19, pp. 315–354, 2003.

[26] E. Choi, M. T. Bahadori, A. Schuetz, W. F. Stewart, and J. Sun, "Doctor ai: Predicting clinical events via recurrent neural networks," in *Machine Learning for Healthcare Conference*, 2016, pp. 301-318.

[27] A. W. Whitney, "A direct method of nonparametric measurement selection," *IEEE Transactions on Computers*, vol. 100, no. 9, pp. 1100-1103, 1971.

[28] T. Cover, and P. Hart, "Nearest neighbor pattern classification," *IEEE transactions on information theory*, vol. 13, no. 1, pp. 21-27, 1967.

[29] C. Sundar, M. Chitradevi, and G. Geetharamani, "Classification of cardiotocogram data using neural network based machine learning technique," *International Journal of Computer Applications*, vol. 47, no. 14, 2012.

[30] F. Pedregosa, et al., ”Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research,* vol. 12, pp. 2825-2830, 2011.

[31] X. Wu, et al., “Top 10 algorithms in data mining,” *Knowledge and Information Systems,* vol. 14, no. 1, pp. 1-37, 2008.

[32] D. J. Hand and K. Yu, “Idiot's Bayes---not so stupid after all?,” *International Statistical Review*, vol. 69, no. 3, pp. 385-398, 2001.

[33] H. Zhang. “The optimality of naive Bayes,” in *Proceedings of the 17th International Florida Artificial Intelligence Research Society Conference (FLAIRS '04). AAAI*, 2004, pp. 562-567.

[34] C. Cortes and V. Vapnik, “Support-Vector Networks”, *Machine Learning*, vol. 20 no. 3, pp. 273-297, 1995.

[35] S. Abe, Support Vector Machines for Pattern Classification, Springer Publishing Company, Incorporated, 2012.

[36] Y. Freund and R. E. Schapire, “Experiments with a new boosting algorithm,” in *Proceedings of the 13th International Conference on Machine Learning (ICML '96)*, 1996, pp. 148-156.

[37] R. E. Schapire, “A brief introduction to boosting,” in *Proceedings of the 16th international joint conference on Artificial intelligence*, 1999, pp.1401-1406.

[38] J. R. Quinlan, C4.5: programs for machine learning, Morgan Kaufmann Publishers Inc., San Francisco, CA, 1993.

[39] T. K. Ho, “Random decision forests,” in *Document analysis and recognition, 1995., proceedings of the third international conference on* (Vol. 1, pp. 278-282), 1995.