Received March 9, 2021, accepted March 29, 2021, date of publication April 2, 2021, date of current version April 16, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3070618

AEP-DLA: Adverse Event Prediction in Hospitalized Adult Patients Using Deep Learning Algorithms

CHIEH-LIANG WU[1,2,3], MING-JU WU[4,5,6], LUN-CHI CHEN[7], YING-CHIH LO[8], CHIEN-CHUNG HUANG[7,9], HSIU-HUI YU[10], MAYURESH SUNIL PARDESHI[11], WIN-TSUNG LO[7,11], AND RUEY-KAI SHEU[7]

[1]Department of Critical Care Medicine, Taichung Veterans General Hospital, Taichung 407, Taiwan
[2]Department of Automatic Control Engineering, Feng Chia University, Taichung 40724, Taiwan
[3]Department of Industrial Engineering and Enterprise Information, Tunghai University, Taichung 407224, Taiwan
[4]Division of Nephrology, Department of Internal Medicine, Taichung Veterans General Hospital, Taichung 407, Taiwan
[5]School of Medicine, Chung Shan Medical University, Taichung 40201, Taiwan
[6]Rong Hsing Research Center for Translational Medicine, College of Life Science, Institute of Biomedical Science, National Chung Hsing University, Taichung 40227, Taiwan
[7]Department of Computer Science, Tunghai University, Taichung 407224, Taiwan
[8]Center of Quality Management, Taichung Veterans General Hospital, Taichung 407, Taiwan
[9]Computer and Communication Center, Taichung Veterans General Hospital, Taichung 407, Taiwan
[10]Department of Nursing, Taichung Veterans General Hospital, Taichung 407, Taiwan
[11]AI Center, Tunghai University, Taichung 407224, Taiwan

Corresponding author: Ruey-Kai Sheu (rickysheu@thu.edu.tw)

This work was supported in part by Tunghai University, Taichung, Taiwan, under Grant MOST 109-2321-B-075A-001, and in part by TVCGH, Taichung, Taiwan, under Grant MOST 108-2314-B-075A-003-MY3.

ABSTRACT Early prediction of clinical deterioration such as adverse events (AEs), improves patient safety. National Early Warning Score (NEWS) is widely used to predict AEs based on the aggregation of 6 physiological parameters. We took the same parameters as the features for AE prediction using deep learning algorithms (AEP-DLA) among hospitalized adult patients. The aim of this study is to get better performance than traditional naïve mathematical calculations by introducing novel vital sign data preprocessing schemes. We retrospectively collected the data from our electronic medical record data warehouse (2007 $\sim$ 2017). AE rate of all 99,861 admissions was 6.2%. The dataset was divided into training and testing datasets from 2007-2015 and 2016-2017 respectively. In real-life clinical care, physiological parameters were not recorded every hour and missed frequently, for example, Glasgow Coma Scale (GCS). The expert domain suggested that missed GCS was rated as 15. We took two strategies (stack series records and align by hour) in the data preprocessing and tripling the values of negative samples for class balancing (CB). We used the last 28 hours' serial data to predict AEs 3 hours later with Random Forest, XGBoost, Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN). It is shown that CNN with CB and align by hour got the best results comparing to the other methods. The precision, recall and area under curve were 0.841, 0.928 and 0.995 respectively. The performance of the model is also better than those proposed in the published literatures.

INDEX TERMS Adverse event (AE), early deterioration indication, early warning scores, electronic medical record, risk stratification.

I. INTRODUCTION

The Institute of Medicine's "To Err Is Human" has been published since 1999. After two decades, David W. Bates and Hardeep Singh reported the progress of patient's safety and pointed out that health information technology (HIT) can help prevent many types of patient safety errors [1]. However, HIT also introduces new problems, including ensuring the safety of the technology itself; the safe use of the technology by clinicians, staff members, and patients; and the effective use of it to improve patient safety. Early warning system (EWS), one of clinical decision support, is used to recognize

The associate editor coordinating the review of this manuscript and approving it for publication was Li He.

the patients with the risk of clinical deterioration and then to trigger aggressive actions to prevent adverse events (AEs). EWS at first was calculated manually. With the advances in electronic health record system, EWS has been embedded into Health information system clinically [2] and decreased the mortality successfully [3].

Recently, two literatures advised that many studies of early warning scores were found to have methodological weaknesses are by Goldstein *et al.* [2] and Gerry *et al.* [4]. Both pointed out the challenge of missing data. Many of the studies did not specified the method of handling missing data and how to impute them. Some studies only used the cases with complete data. In real-life clinical care, physiological parameters are not recorded every hour and are missed frequently. These will reduce the accuracy of HIT-based EWS in predicting clinical deterioration and also challenge the deployment clinically.

In this study, we retrospectively collected the cohort data of electronic health record (EHR) from 2007 to 2017 at Taichung Veterans General Hospital, Taiwan. We collected the parameters from National Early Waring Score (NEWS 2 [5], [6]), which is standardizing the assessment of acute-illness severity in the NHS. We have used the last 28 hours serial data to predict clinical deterioration 3 hours later with machine learning algorithms including Random Forest, XGBoost with Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN). We took two strategies (stack series records and align by hour) to handle the missing data by the data preprocessing. The study focused on the effect of different methods of managing missing data under the physician as domain expert, not only for the precise prediction but also for easy deployment clinically. The key to get higher performance using deep learning is to have good data preprocessing strategy for vital sign.

A. BACKGROUND KNOWLEDGE

Several machine learning algorithms [7] are present, each of them are categorized based on supervised, unsupervised with classification, regression, clustering and dimension reduction. So in AEP-DLA, we will be predicting numeric accuracy and will be using supervised learning with classification and regression for continuous outcome. Random forest is one of the suitable algorithm that can be used to take average of many decision trees. Here, all trees can perform better only when combined to get overall better performance. Even though it is requiring expertise to understand results but are known for high quality results and fast to train. XGboost algorithm used is also a type of classifier, similar to gradient boosting but with multiple advantage in penalizing, proportional shrinking, newton boosting and extra randomization parameters of decision trees. Neural networks known for exchanging message with interconnected neurons, where deep learning is its one of the structure that uses several layers connected serially. Specially designed to handle complex tasks. Challenge here is to train multiple layers for operations and understand its predictions.

B. APPLICATIONS FOR AEP-DLA CAN BE APPLIED IN VARIOUS SCENARIOS

1) SMART HOSPITAL

Smart hospitals are basically equipped with several continuous and connected health data monitoring devices. Special section for each disease type, critical care units, group of expert doctors, advance surgery and scanning equipment's helps us to get dedicated care and emergency operating on patients.

2) SMART CLINICS

A smart clinic can be used to connect to specialized hospitals. Also results can be communicated live to super-specialty hospitals to get better treatment and health checkup analysis for the doubted critical patients.

3) OLD AGE HOME

Several senior citizens can be found in old age homes. So they can be categorized as per specific diseases and treated accordingly at local. Continuous and fixed interval based monitoring is possible by using smart watch for blood pressure, heart rate, consciousness, etc. Monitoring video based surveillance and consulting is recently made available on large by private hospitals.

4) SPORTS STADIUM CLINIC

Sports players injured or facing unhealthy situations can be monitored and consulted. Data can be collected while shifting to hospital or until the patient gets critical, which is helpful to analyze condition before emergency.

5) SMART AMBULANCE

While the patients are shifted or transferred by travelling to nearby hospital. Smart ambulance can be used to monitor multiple parameters and get temporary medication by the present nurse or doctor. It can also be used to provide checking and remote analyzing of patient's health thus recorded and re-used later.

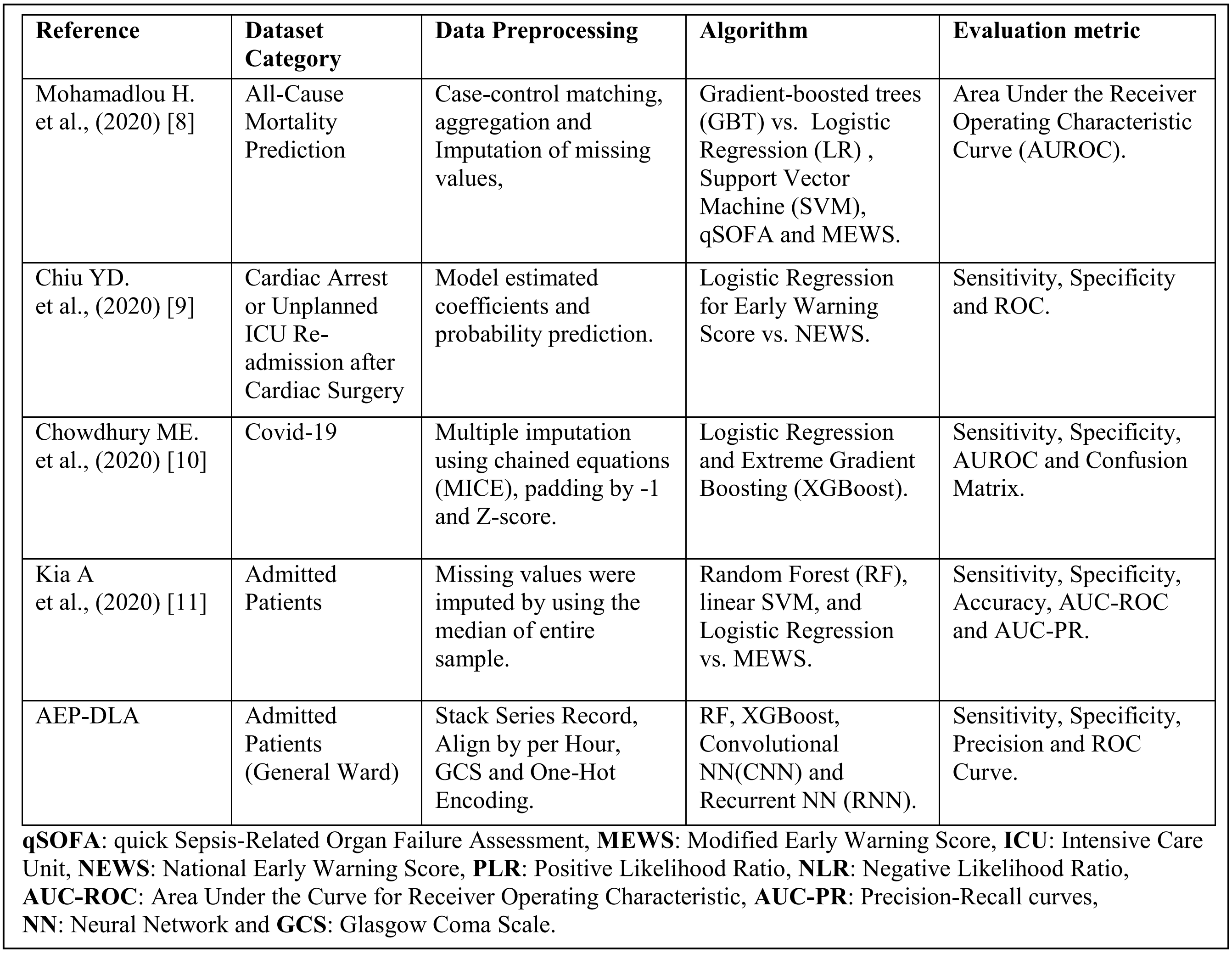
6) REMOTE CENTER CLINIC

Remote clinics in rural areas can be used to collect specific interval data of beginner level patients. Emergency units can be used to monitor health from remote center and get consulting to avoid travelling to avoid travelling time for curable situations.

II. LITERATURE SURVEY

The literature survey presents well known models, which are referenced for this work. The table 1 presents detail comparison with some recent research. A systemic review of early warning score for detecting clinical deterioration and focused on the methodology is conducted by Gerry *et al.* [4]. It concluded that poor methods and inadequate reporting were found in most studies, and all studies were at risk of bias. Methodological problems could result in scoring systems that

TABLE 1. Comparison of AI-based different Early Warning system (EWS) Reference models.



Reference	Dataset Category	Data Preprocessing	Algorithm	Evaluation metric
Mohamadlou H. et al., (2020) [8]	All-Cause Mortality Prediction	Case-control matching, aggregation and Imputation of missing values,	Gradient-boosted trees (GBT) vs. Logistic Regression (LR) , Support Vector Machine (SVM), qSOFA and MEWS.	Area Under the Receiver Operating Characteristic Curve (AUROC).
Chiu YD. et al., (2020) [9]	Cardiac Arrest or Unplanned ICU Re-admission after Cardiac Surgery	Model estimated coefficients and probability prediction.	Logistic Regression for Early Warning Score vs. NEWS.	Sensitivity, Specificity and ROC.
Chowdhury ME. et al., (2020) [10]	Covid-19	Multiple imputation using chained equations (MICE), padding by -1 and Z-score.	Logistic Regression and Extreme Gradient Boosting (XGBoost).	Sensitivity, Specificity, AUROC and Confusion Matrix.
Kia A et al., (2020) [11]	Admitted Patients	Missing values were imputed by using the median of entire sample.	Random Forest (RF), linear SVM, and Logistic Regression vs. MEWS.	Sensitivity, Specificity, Accuracy, AUC-ROC and AUC-PR.
AEP-DLA	Admitted Patients (General Ward)	Stack Series Record, Align by per Hour, GCS and One-Hot Encoding.	RF, XGBoost, Convolutional NN(CNN) and Recurrent NN (RNN).	Sensitivity, Specificity, Precision and ROC Curve.

qSOFA: quick Sepsis-Related Organ Failure Assessment, **MEWS**: Modified Early Warning Score, **ICU**: Intensive Care Unit, **NEWS**: National Early Warning Score, **PLR**: Positive Likelihood Ratio, **NLR**: Negative Likelihood Ratio, **AUC-ROC**: Area Under the Curve for Receiver Operating Characteristic, **AUC-PR**: Precision-Recall curves, **NN**: Neural Network and **GCS**: Glasgow Coma Scale.

perform poorly in clinical practice, which might have detrimental effects on patient care. One of their recommendations were multiple imputation is the best practice approach for accounting of missing data in the analysis.

National Early Warning Score (NEWS) by Royal College of Physicians [5], [6] was developed for adult's patients to detect the clinical deterioration with their physiological parameters that are part of routine measurements in NEWS. A final score is calculated based on aggregation of data by parameter weighting of various physiological parameters. The monitoring frequency of patients, the response to clinical urgency based on triggers and escalations of care levels are used for acute illness severity features by scaled response. A standardized chart is developed for recording the routine patient checkup of NEWS parameters with a supported online training implementation for analysis. NEWS has been widely used with or without minor modification in many hospitals. Machine learning used in intensive care unit (ICU) for circulatory failure for early prediction is presented by Hyland SL *et al.* [6]. The data preprocessing was performed using artifact removal, category-based variable merging with adaptive imputation and state annotation.

High feature importance variables were used to be classified by gradient boosting classifier which is evaluated by specificity, precision, recall and frequency. Artificial intelligence for patient's health deterioration detection in rapid response system is demonstrated by Cho KJ *et al.* [12]. The dataset consists of vital signs for predicting in-hospital cardiac arrest and ICU admission, which is classified using RNN, long-short term memory (LSTM), rectified linear unit (ReLU), four fully connected layers and softmax for binary (0,1) evaluation. The results are presented as sensitivity, specificity prediction and ROC curve. Explainable artificial intelligence (xAI) for evaluation EHR records used in predicting acute critical illness is presented by Lauritsen SM *et al.* [13]. AI models showed trade-off between sensitivity and specificity, so to overcome it xAI models was constructed and data from disease category is analyzed using AUROC and AUPRC. The xAI model showed better performance than SOFA, MEWS and Gradient boosting vital signs.

AI for Covid-19 prognostic modelling in UK is demonstrated by Abdulaal A *et al.* [14]. A mortality risk scoring system from hospital admission using artificial neural network (ANN) is developed with hyper-parameter tuning with importance from Shapley additive explanations (SHAP) values. The evaluation is performed using k-fold cross-validation, (training and loss) vs. epoch and AUROC. AI in critical care prediction to be used during prehospitalization services is presented by Kang DY *et al.* [15]. The data is normalized by z-score and given as input to feedforward network with adam optimizer and tensor-flow as backend. The ROC curve outperformed NEWS, MEWS, emergency severity index (ESI) and Korean triage and acuity system (KTAS). Machine learning applied in EWS of Cardiac Arrest is demonstrated by Chang HK *et al.* [16]. The feature selection is performed of CPR as well as on non-CPR patients using sequential forward selection and applied to decision trees and random forest to higher ROC curve validation.

A review on EWS scores for patient's clinical signs deterioration is presented by Smith *et al.* [17]. A systematically review of 21 articles is present and concluded that early warning system tools perform well for predicting death and cardiac arrest within 48 hours but the impact on in-hospital health outcomes and utilization of resources remains uncertain, owing to the methodological limitations. The Modified Early Warning Score (MEWS) by Galen *et al.* [18] is used for recognizing hospitalized patient's clinical deterioration by performing an analysis of real life settings to MEWS protocol adherence, measuring MEWS daily by predictive value determination for Serious Adverse Events (SAEs): ICU admissions and readmissions, cardiac arrests and death. A criteria score is a setup to compare across 6 different wards of hospitalized patients by follow-up of 30-day to be compared presence and absence of critical score. It suggested that MEWS needed modified according to different diseases. Early Deterioration Indicator (EDI) by Ghosh *et al.* [19] Mortality and ICU level of care can be reduced by keeping patients in general wards by having timely interventions. EDI uses continuous risk scores to be calculated from vital signs log likelihood risk. EDI was validated by comparing it with MEWS and NEWS, which is trained by using data mining knowledge of large datasets. However, the missing data was a problem in constructing EDI model. Churpek *et al.* [20] created the organ failure assessment related to quick sepsis (qSOFA) score, it is to identify patients with high risk not within ICU. It is used for clinical deterioration detection compared with EWS and systematic inflammatory response syndrome (SIRS). It concluded that EWS is more accurate than qSOFA in non-ICU patients.

Electronic Cardiac Arrest Risk Triage (eCART), NEWS and MEWS scores for ward patient's health deterioration are compared by Green *et al.* [21]. It can help in providing immediate attention to critical patients by using electronic algorithms. The study considers data from five different hospitals in the US from 2008-2013. Predicting for patient's ICU transfer, cardiac arrests and death up to 24 hours' observation as the composite outcome is found to be better and accurate on eCART than other paper-based observations based on AUROC. Sepsis prediction in the ICU using machine learning by Desautels *et al.* [22] uses minimal variable set for prediction and is compared with existing scoring system for performance including data sparsity investigation. The data preprocessing is missing values are imputed by carry-forward subsequent values bin and for different data back-fill from first subsequent bin. Multi-parameter Intelligent Monitoring in Intensive Care (MIMIC)-III dataset is used to predicting sepsis based on patient data in ICU using classification in machine learning. This classification is compared with recent scoring systems i.e. qSOFA, MEWS, SIRS, SOFA, SAPS-II for acquiring septic prediction and is found to be performing well even with random missing data evaluated by AUROC. Wellner *et al.* [23] proposed a machine learning approach for unplanned transfers to ICU prediction. The data is taken from three children hospitals that are used to check for predicting performance for unplanned ICU transfers by using different predictor variables. Different training and testing data were used with the cases from suspects of meeting within single/multiple five criteria of unplanned transfers and for those transferred to the ICU from floor. Neural networks and logistic regression were used for classification models and 1-16 hours of horizon prediction was used for modeling performance evaluation. Accuracy was determined in advance even before 16 hours of patient's deterioration by AUROC. Clinical deterioration prediction using conventional regression and machine learning methods of multicenter comparison by Churpek *et al.* [24] presents ML dominates conventional regression. The data pre-processing takes missing values from prior blocks, in case of no previous blocks them median values are imputed. The different techniques in ML are used to predict survival analysis using discrete time by using health parameters for predicting the outcome. Different training and testing data were used in which, the random forest was found to be more accurate than others with MEWS AUC, whereas spline prediction logistic regression AUC was more accurate than other regression models. Concluding that improved identification is achieved for critical patients. Machine learning for hemodialysis patients' quality of life (QOL) prediction is presented by Saadat *et al.* [25] uses algorithms based on naïve bayes and classification trees. The classification tree was found performing better with AUC while considering environmental and psychological domains for QOL.

The limitations of previous papers that are studied in the above survey are as follows:

1. Insufficient data preprocessing techniques are used to handle missing data.
2. The features (measurement of physiological parameters) are different partly among those studies.
3. The time frame of event and non-event data (with and without clinical deterioration) for constructing model are varied among those studies.
4. Traditional statistic study design achieved good power in predicting clinical deterioration. Machine learning

TABLE 2. Various features collected or provided by machine settings to patient for the purpose of data collection provided as input to the adverse event predictor.

Feature Type	Unit (Measurement)
Breathing	/min
Pulse	/min
Body Temperature	°C
Systolic Blood Pressure	mmHg
Diastolic Blood Pressure	mmHg
Oxygen Use	Y/N
VS_HR_PreEvent	Hours
Oxygen Saturation	%
Subspecialty	Name
Glasgow Coma Scale • Eye Opening • Verbal Response • Motor Response	Y/N (range)

or deep learning probably construct better algorithm to predict clinical deterioration precisely.

A deep learning algorithms based method is designed for our AEP-DLA. Our goal is to include the features of NEWS and increase the accuracy of AEP-DLA. The paper is further organized in the following manner: Section III. The methodology will be focusing on the system model and algorithms used for machine learning and deep learning. Section IV. Experiments will be discussing data preprocessing and the performance of various algorithms with its comparison. In the end, we will have Conclusion followed by Acknowledgement and References.

III. METHODOLOGY

1) Settings: The study was done in Taichung Veterans General Hospital (TCVGH), a 1500-bed academic hospital in central Taiwan. The ethical committee/institutional review board (Institutional Review Board (II)107-B-08 Board Meeting) approved the study protocol (protocol no./IRB TCVGH No: CE18209B). Therefore, written informed consent from the participants was waived. Patients information was anonymized and de-identified prior to analysis in the study.

2) Patients AND Data Retrieving: The enrolled criteria were (1) patients hospitalized at general wards (2) age $\geq$ 20 years old. The exclusion criteria were those patients with one of the following: (1) hospitalization day less than one day ($\leq$ 24 hours), and (2) direct admission to ICU, and (3) the patients who have had artificial airways at admission. We retrospectively collected the data from our electronic medical record (EMR) data warehouse 2007/01/01∼2017/12/31.

3) Definition of Adverse Event: The deteriorating patients were grouped as "adverse event (AE)" if they received cardiopulmonary resuscitation, were transferred to ICU with unexpected deterioration, or died. The other patients were regarded as "no adverse event (NAE)". Those patients with scheduled admission to ICU after surgery were regarded as NAE. Only the first episode of AE was studied.

4) Vital Sign Data: Vital sign data measured in various frequencies according to clinical needs. In a specific timing with no measured data, we get the previous closest one as the representative. The data collected from the patient or provided with facilities of supporting life care can be categorized as listed in Table 2.

The method for our model is shown in Figure 1. As shown in Figure 1. The AEP-DLA System Model, it is basically divided into four parts: Input data as vital signs, data preprocessing, training model and emergency warning system predictor. Once the vital signs are collected, the data would be cleaned first, so that it becomes suitable for data preprocessing. We define data cleaning separately as it is used to identify and correct improper records acquired as input. The data which is considered to be inaccurate is then can be either replaced, modified or deleted as per the criteria. The purpose of such data cleaning operation is to make dataset consistent to be processed by the predictor system and hence valid. The accuracy within the data can be caused due to several issues related to entry errors, data corruption, transmission errors, etc. It can also involve harmonizing data or standardizing them which relating terms to short codes and vice versa respectively. Data cleaning is a basic requirement done before data preprocessing as the later part involves validation instead of accuracy for processing. VS_HR_PreEvent is used to indicate all feature's time limits before adverse events. Whereas, the subspecialty indicated the section for the disease diagnosis concerning the disorders of the specific organ.

In the following section, we are going to present the working model of adverse event prediction using machine and deep learning algorithms i.e. Random Forest, XGBoost, CNN and RNN. The input to the working model is data collected from each patient in the hospital ward, which is then preprocessed and applied to deep learning algorithms for evaluating results.

A. INPUT DATA FOR VITAL SIGNS

The input provided to the model is the series record as shown in the bottom left of the system model. The series data presents x-axis as the data collected from different patients and y-axis as per the time data is recorded in hours. Each data collected or provided by machine settings to the patient is shown from last 28 hours of scale, as we are using an algorithm prediction, we need past records for its input.

B. DATA PREPROCESSING

The input provided to data preprocessing is in the form of cleaned data. Data preprocessing is basically required to provide data validation. As it is an important step in machine learning, it is used to check whether the information gathering was loosely acquired that can collect out-of-range values, missing values and may later on lead to improper data combinations, etc. Redundant data can also be used to

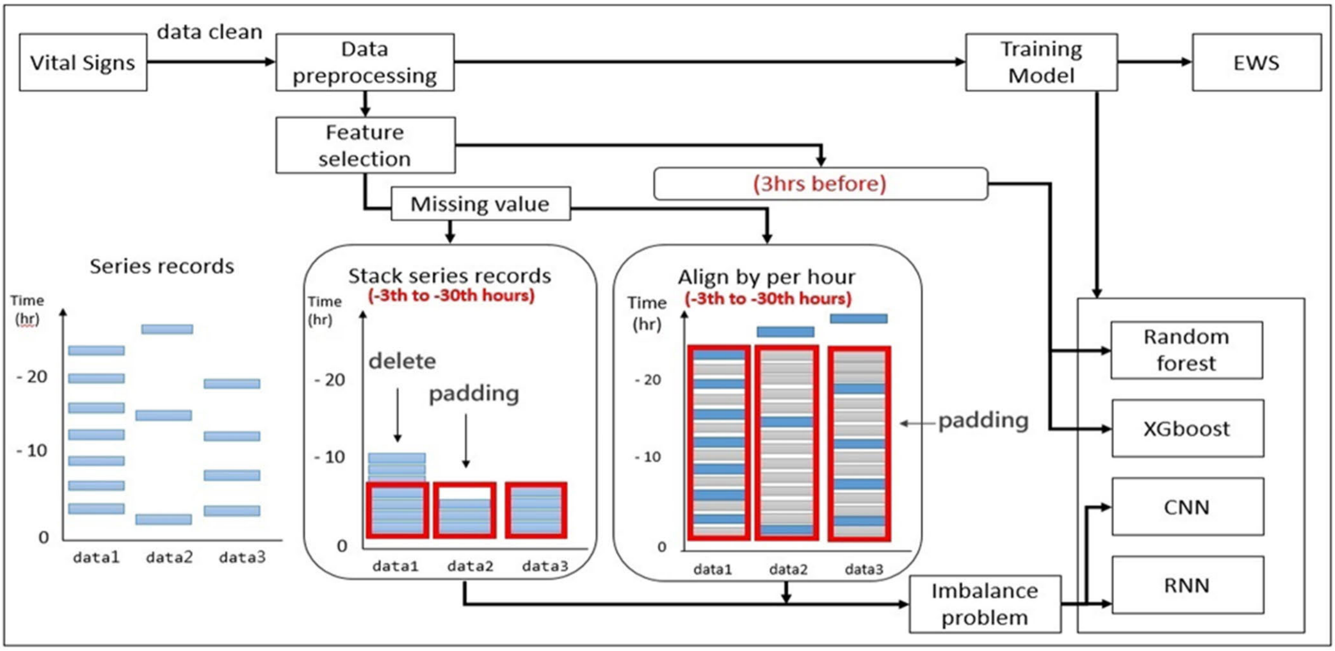


FIGURE 1. The AEP-DLA system model.

overcome using preprocessing, data quality is considered to be an important factor for AEP.

Data preprocessing can be viewed broadly by cleaning of the data first, selecting of an appropriate instance, normalizing the data for smooth processing, transforming the data from original to the required form, extracting features from the acquired data and selecting the most appropriate data values, etc. Henceforth data preprocessing is considered to be a crucial part of final interpretation, as the outcome can be affected due to it. The feature selection done here is as presented in Table 2. Once the features are captured we either check for recovering the missing values or we directly provide the patients data produced 3 hours before of the current time. This paper proposes two strategies in the data preprocessing stage to deal with the problem of missing values, which are stack series records and align by per hour method. To predict the early warning signs after 24 hours, the algorithm requires a total of 28 records from the vital signs data during last 3 to 30 hours. If we have patients vital sign data value of one record from only last 3 hours with 15 features as seen before in Table 2, consisting of 12 features + GCS replaced by total of 1 feature + One Hot encoding replaced as 6 features to predict the early warning sign, which is then can be applied on the machine learning algorithm of random forest or XGBoost. If we have patients vital sign data from last 3 to 30 hours with 15 features of 28 records to predict early warning signs, then we can use deep learning algorithms CNN without class balance, CNN with class balance, alignment based(/hour) CNN (with class balance) and alignment based(/hour) RNN (with class balance).

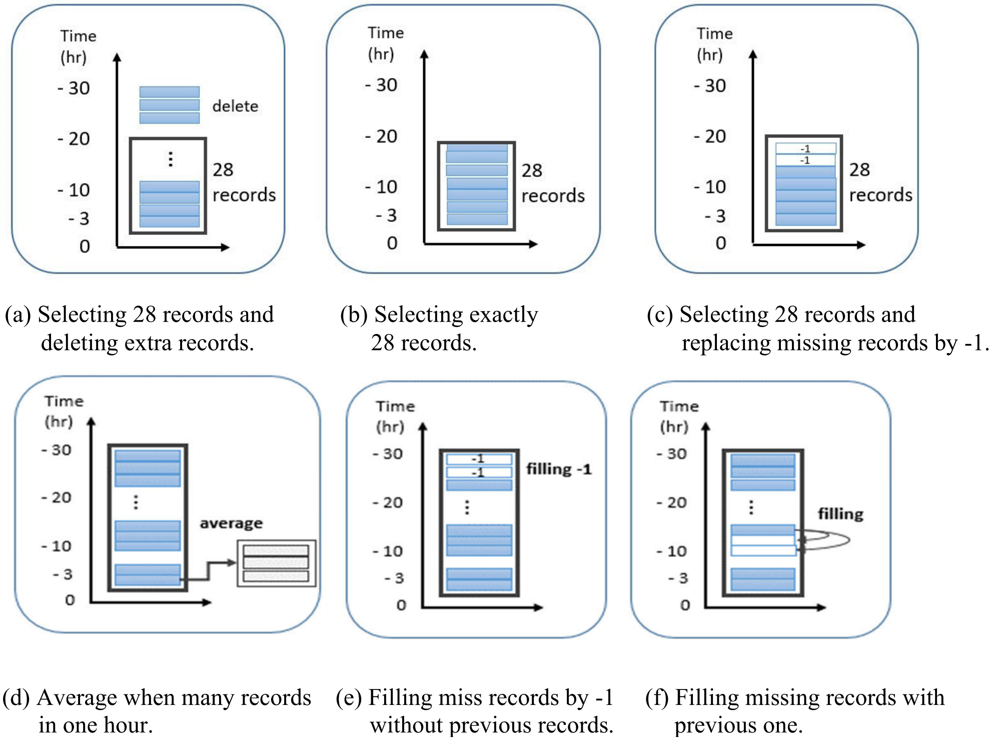
The missing values in data preprocessing can be recovered by the following methods as stated below:

1) STACK SERIES RECORD

As shown in the Figure 1 and 2, three patients data are available i.e. data 1, data 2 and data 3. In this scenario, the data 1 is stacked from the series record data 1. After stacking, data 1 has exceeded the limit of 28 records, hence we delete up to the threshold of 28 records, as we require only 28 records for processing. Data 2, it levels up to the required 28 records, hence no operation is required to be applied here. Whereas for data 3 when stacked from series record, is having not enough records to reach the threshold of equaling 28 records, so it's an underflow, hence we do padding for this data 3 and is filled by -1 value.

2) ALIGN BY PER HOUR

As shown in Figure 1 and 2, the series records are kept as it is and instead of stacking, only padding is performed in this method. So for the vital sign of patient data 1, we can see that blue bars represents recorded data and the grey bars for padded data. When multiple data is recorded in one hour then it is average to single data per hour. In case of data 2, padding is done by -1 value, when the previous data record does not exist for a long time, which is less than the last 30 hours. In case of data 3, when the data is missing in the middle part then padding is done with recent record. In short whenever any operation is performed for aligning then data is always searched in forward of time. Here, data is only considered from last 3 to 30 hours. In short, the data recorded currently is also not required or can be deleted for the prediction from current time. When previous results are present then we can consider doing padding of data recent records until new records are obtained. Therefore, the record of last 3 to



(a) Selecting 28 records and deleting extra records.

(b) Selecting exactly 28 records.

(c) Selecting 28 records and replacing missing records by -1.

(d) Average when many records in one hour.

(e) Filling miss records by -1 without previous records.

(f) Filling missing records with previous one.

FIGURE 2. Handling missing values by I. Stack series records method (a, b, c), and II. Alignment by Per Hour Method (d, e and f).

30 hours can thus be obtained by using such method, in case of missing values.

3) GLASGOW COMA SCALE

The Glasgow coma scale (GCS) is used to represent three features of the patient's health, which can be considered as an initial parameter. Eye opening (1-4), verbal response (1-5) and motor response (1-6) are the range values for evaluating their well-being on the scale. The higher the values, the better is the test response. In case of input to the machine learning/deep learning model, when GCS values are found to be missing then they are rated as a total of 15.0 scale as found to be considered as normal, instead of rating them all separately.

4) CATEGORY VALUE

Missing category values in the dataset are replaced with a binary class matrix by one-hot encoding. The process of converting categorical variables into suitable input for the machine learning algorithm form, resulting in a better prediction by one-hot encoding. So to avoid label encoding problem of considering higher value for category as a superior category, the one-hot encoding thus operates by doing category binarization by having it as a training model feature.

C. ALGORITHMS

In data preprocessing, the data performs both preprocessing on Glasgow Coma Scale and Category Value. However, Stack Series Record and Alignment by Per Hour are different kinds of data preprocessing methods. Both kinds of preprocessing methods had been experimented and evaluated within the experiment section. We also consider continuous data normalization by using min-max. Min-max is basically used to normalize all numeric range values to between 0 and 1 by using feature scaling. It is also known as to be unity-based normalization. When there are two arbitrary points in the dataset a and b, then the value restriction can be achieved by its generalization. The min-max(X') can be expressed as given in equation 1:

$$X' = a + \frac{(X - X_{\min})\,(b - a)}{X_{max} - X_{\min}} \tag{1}$$

The above algorithm 1 pseudo-code presents the data preprocessing performed on the patient's records using stack series method. In step 1, the raw data(*DRaw*) collected as patient's health record is given as input to the algorithm. In step 2, the stack size *(Stack$_{size}$)* is input which is required while preprocessing. In step 3, processed data *(DProcessed$_{id}$),* a stack is used to store output by the algorithm. In step 4, *DProcessed* stack is initialized to NULL for storing output. In step 5, the *Stack$_{size}$* is declared as the part of input from step 2. In step 6, the first IF condition checks whether the particular patients raw record data *DRaw$_{id}$* is greater than the *Stack$_{size}$*? Then in step 7, the extra records greater than the size are deleted from *DRaw$_{id}$* and stored in *DProcessed$_{id}$*. In step 8, Else-IF is used to check whether *DRaw$_{id}$* is less than the *Stack$_{size}$*? Then in step 9, padding by -1 is applied to fill the values upto stack size. In step 10,

Algorithm 1 Data Preprocessing by Stack Series Record Method

```
1.  Input:   $DRaw$, Patient's Raw Data Record
2.           $Stack_{size,}$ Stack Size for Preprocessing
3.  Output:  $DProcessed_{id}$,, Preprocessed Patient's
                 Data Records
4.  $DProcessed = \emptyset$
5.  Declare $Stack_{size}$
6.  If $DRaw_{id} > Stack_{size}$
7.      $DProcessed_{id} =$ Delete Extra Records($DRaw_{id}$)
8.  Else if $DRaw_{id} < Stack_{size}$
9.      $DProcessed_{id} =$ Padding($DRaw_{id}$)
10. Else
11.     $DProcessed_{id} = DRaw_{id}$
12. Return $DProcessed_{id}$
```

Algorithm 2 Data Preprocessing as Alignment by Per-Hour Record Method

```
1.  Input:   $DRaw_{PH}$, Patient's Per-Hour Record Raw
               Data
2.           $Stack_{Top,}$ Stack Size for Preprocessing
3.           $Block_{Size,}$ Block Size in Stack for
               Preprocessing
4.  Output:  $DProcessed$, Preprocessed Patient's
               Records Data
5.  $DProcessed = \emptyset$
6.  Declare $Stack_{Top}, Block_{Size}$
7.  If $DRaw_{ID,PH} > Block_{Size}$
8.      $DProcessed_{ID,PH} =$ Average($DRaw_{ID,PH}$)
9.  Else if $DRaw_{ID,PH} ==$ NULL
            and $Stack_{Top}(DRaw_{ID,PH}) ==$ True
10.     $DProcessed_{ID,PH} =$ Padding($DRaw_{ID,PH}$)
11. Else if $DRaw_{ID,PH} ==$ NULL
            and $Stack_{Top}(DRaw_{ID,PH}) ==$ False
12.     $DProcessed_{ID,PH} =$ Replicate($DRaw_{ID+1,PH}$)
13. Else
14.     $DProcessed_{ID,PH} = DRaw_{ID,PH}$
15. Return $DProcessed_{ID,PH}$
```

the Else condition is used to replicate the data to $DRaw_{id}$ from $DProcessed_{id,}$ as they have same stack size in step 11. In step 12, the pre-processed $DProcessed_{id}$ is returned by the algorithm.

The algorithm 2 pseudo-code presents the data preprocessing performed on the patient's records using alignment by per hour record method. In step 1, the input given to the algorithm is raw patient's per hour record data($DRaw_{PH}$). In step 2, the input is top of the stack($Stack_{Top}$) value. In step 3, the input is fixed block size in each stack($Block_{Size}$). In step 4, the output is given as preprocessed patient's record data (*DProcessed*). In step 5, the processed data stack is initialized to NULL. In step 6, the stack top and block size needs to be declared, as a part of input. In step 7, the IF condition checks that whether the per hour patient's raw record data is recorded multiple times, which is greater than block size($Block_{Size}$)? If true then in step 8, the data is averaged and stored as single hour data for that particular patient's ID($DRaw_{ID,PH}$) in processed data ($DProcessed_{ID,PH}$). In step 9, Else-If checks whether the per hour raw data of a patient is NULL and that per hour raw data is at top of the stack($Stack_{Top}$)? then in step 10, padding by -1 is applied to that raw record and is stored in that respective preprocessed data ($DProcessed_{ID,PH}$). In step 11, Else-If checks whether the per hour raw data of a patient is NULL and that per hour raw data is not at top of the stack($Stack_{Top}$) but lower than stack top? then in step 12, the per hour record present above the current record in the stack is replicated to the current record block and is updated in that particular patient's preprocessed record ($DProcessed_{ID,PH}$). In step 13, else no above conditions are matched then that raw record($DRaw_{ID,PH}$) is assumed to be valid and in step 14 is added directly to per hour preprocessed record ($DProcessed_{ID,PH}$). In step 15, the per hour pre-processed records($DProcessed_{ID,PH}$) is returned by the algorithm.

The algorithm 3 presents the final adverse event prediction (AEP) pseudocode. The AEP algorithm combines the two data preprocessing for stack series records and align by per hour method. In step 1, input required by the algorithm is vital signs of patient, which are recorded as a part of hospitalization process. In step 2, threshold determines the limit or the boundary after which alert needs to be raised by the AEP system. In step 3, the score is the output generated by the AEP system at the end using the best score from either the machine learning or deep learning algorithms. In step 4, the alert is raised as a warning to indicate approaching adverse event of the patient. In step 5, multiple local variables initialized are raw data(*DRaw*), processed data(*DProcessed*), cleaned data(*DCleaned*), candidate score(*CandidateScore*), machine learning/deep learning *Score* and *Message*. In step 6, the data is cleaned for vital signs by checking whether there are any missing values? In step 7, the IF condition checks is data cleaned (*DCleaned*) true? In step 8, if data cleaned is true then model is trained from machine learning and deep learning to get the best score. In step 9, Else condition is raised for missing values then vital sign data(*DCleaned*) is selected which checks for more details by feature selection including NULL values, normalizing, extraction, transformation, etc. later stored in data raw($DRaw_{ID}$) with the identity of a particular patient in step 10. In step 11, If the raw data($DRaw_{ID}$) required to be processed is less than or equal to last 3 hours then in step 12, random forest algorithm is applied and its result are stored in *CandidateScore1*. In step 13, again XGBoost algorithm is applied on the same raw data($DRaw_{ID}$) and results stored in *CandidateScore2*. In step 14, best score from *CandidateScore1* and *CandidateScore2* is selected to be stored in *Score*. In step 15, Else-If check whether the recorded data is from last 3 hours to last 30 hours of raw data($DRaw_{ID}$) then in step 16 stack series data preprocessing algorithm is applied and stored as preprocessed data($DProcessed_{Series}$). In the similar way, in step 17, align by

Algorithm 3 Adverse Event Prediction

```
1.  Input:   VitalSigns, Vital Signs of Patient
2.           Threshold, To Determine Adverse Event Prediction Limit
3.  Output:  Score, Score for Selecting the Best Training Data
4.           Alert, Early Warning System Alert by AEP-DLA
5.  (DRaw, DProcessed, DCleaned, CandidateScore, Score, Message) = $\emptyset$
6.  DCleaned = Data Cleaning(VitalSigns)
7.  If DCleaned == True
8.      Score= Model-Train(DCleaned)
9.  Else
10.     $DRaw_{ID}$ = Feature Selection(DCleaned)
11.     If TimeStamp($DRaw_{ID}$) <= −3 hours
12.             CandidateScore1 = Random Forest($DRaw_{ID}$)
13.             CandidateScore2 = XGBoost($DRaw_{ID}$)
14.             Score = Best Score(CandidateScore1, CandidateScore2)
15.     Else If TimeStamp($DRaw_{ID}$) > −3 hours and TimeStamp($DRaw_{ID}$) <= −30 hours
16.             $DProcessed_{Series}$ = Stack Series($DRaw_{ID}$)
17.             $DProcessed_{Aligned}$ = Align by Hour($DRaw_{ID}$)
18.             DProcessed = Class Imbalance ($DProcessed_{Series}$, $DProcessed_{Aligned}$)
19.             CandidateScore3 = Convolutional Neural Network (DProcessed)
20.             CandidateScore4 = Recurrent Neural Network (DProcessed)
21.             Score = Best Score(CandidateScore3, CandidateScore4)
22. If Score <= Threshold
23.     Message = "No Alert"
24. Else
25.     Message = "AEP Alert"
26. Return Message
```

per hour data preprocessing is applied on raw data($DRaw_{ID}$) and is stored in processed data($DProcessed_{Aligned}$). In step 18, class imbalance is solved by balancing the proportion of positive and negative sample approximately and then the preprocessed data is stored in preprocessed(*DProcessed*). In step 19, the convolutional neural network is applied on preprocessed data(*DProcessed*) and the results are stored in *CandidateScore3*. Similarly, in step 20, the recurrent neural network is applied on preprocessed data(*DProcessed*) and the results are stored in *CandidateScore4*. In step 21, the best score from previous two steps is stored in *Score*. In step 22, if the obtained Score from previous any of the results is less than or equal to threshold value then it is considered to be *Safe/No Warning*. In step 24, *Else* when the *Score* is greater, the system raises Adverse Event Prediction(AEP) *Alert*. Finally, the message is returned by the AEP algorithm.

D. MATHEMATICAL ANALYSIS OF THE DEEP LEARNING TRAINING MODELS

1) CONVOLUTIONAL NEURAL NETWORK (CNN)

CNN is known for learning features in hierarchy automatically for the purpose of classification [26], [27]. The feature map constructs, higher layers feature learning by complex, translation and distortion invariant hierarchical approach. A neural network basically consists of perceptron layer *(L+1)* hidden layer with input units *D*, output units' *C* and many hidden units, where units are arranged in layers.

$$y_i^{(l)} = f\left(z_i^{(l)}\right) \; with$$
$$z_i^{(l)} = \sum_{k=1}^{m^{(l-1)}} w_{i,k}^{(l)} y_k^{(l-1)} + w_{i,0}^{(l)} \tag{2}$$

The layer l i$^{\text{th}}$ unit computes the output as given in equation 2, weighted connection k$^{\text{th}}$ to i$^{\text{th}}$ unit in layer l to layer 1 respectively is denoted as $w_{i,k}^{(l)}$, bias which is unit external input is denoted as $w_{i,0}^{(l)}$. We have $C = m^{(L+1)}$ and $D = m^{(0)}$, given as layer L number of units denoted as $m^{(l)}$.

$$\sigma(z) = \frac{1}{1 + \exp(-z)} \tag{3}$$

The sigmoid activation function σ as given in equation 3 represents high dimensional network with non-linear properties having slow convergence. It is basically mapping of complex functions between response variables and input z. So the input times weight is added with bias and activation.

$$T_x := \{(x_n, t_n) : 1 \le n \le N \tag{4}$$

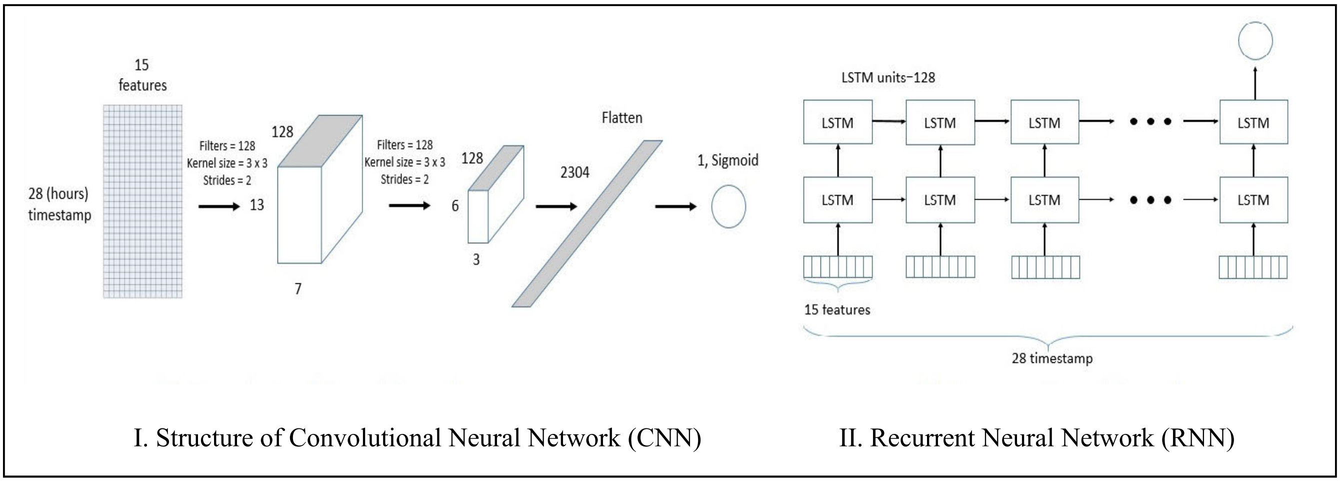
Network weights determined by target specific mapping approximations g is done by the supervised training. The training data set gives mapping due to the unknown g practically. The set of training is given in equation 4, where x_n gives input value and $t_n \approx g(x_n)$ is the output value, possibly noise.

$$-\frac{1}{\sqrt{m}^{(l-1)}} < w_{i,j}^{(l)} < -\frac{1}{\sqrt{m}^{(l-1)}} \tag{5}$$

The weight initialization w is crucial for technique of iterative optimization. The weights in equation 5, are chosen randomly in that range. Each unit input distribution are based using the assumptions by Gaussian distribution and unity order approximation is the actual input ensured. Here, we can have optimal learning by using activation function of logistic sigmoid.

$$MSE = \frac{\sum_{i=1}^{n} (y_i - y_i^p)^2}{\text{n}} \tag{6}$$

The objective function of minimizing is used here, also known as loss function used for measuring the predicting outcome. The Mean Square Error (MSE) or Quadratic loss belongs to the type of most commonly used regression loss



I. Structure of Convolutional Neural Network (CNN)

II. Recurrent Neural Network (RNN)

FIGURE 3. **Neural network learning models.**

TABLE 3. **CNN Hyperparameter configuration.**

Layer Input	Kernel Size	Stride	Number of Filters	Output Shape
Conv 1	(3x3)	(2x2)	128	(13x7x128)
Conv 2	(3x3)	(2x2)	128	(6x3x128)
FC	-	-	2	(2,)

category. MSE is represented as in equation 6, as the sum of squared distances within the given target variable and predicted values.

The Table 3. shows hyperparameter configuration for CNN Figure 3. I. CNN, which provides optimal settings for achieving better output with convolutional layer 1 (Conv 1), convolutional layer 2 (Conv 2) and fully connected layer (FC) having a flatten output of 2304 with sigmoid as the final activation function.

2) RECURRENT NEURAL NETWORK (RNN)

Connectionist models RNNs are used collect sequence dynamics by network node cycles [28]. It is basically use to collect sequence state from a large context window. LSTM allows to train, optimize for achieving large scale learning.

$$h^{(t)} = \sigma(W^{hx}x^{(t)} + W^{hh}h^{(t-1)} + b_h) \tag{7}$$

$$\hat{y}^{(t)} = softmax(W^{yh}h^{(t)} + b_y) \tag{8}$$

Recurrent edge nodes known for connecting adjacent time steps receive $x^{(t)}$ as current input from data point, $h^{(t-1)}$ value from previous state hidden node and $\hat{y}^{(t)}$ is given as output. In equation 14, the weights within the input and hidden layer is given as W^{hx} matrix, whereas weights within hidden layer and recursively in time steps adjacently to itself is given as W^{hh}. The offset is learned by each node using the bias parameter b_h and b_y vectors in equation 7 and 8 respectively.

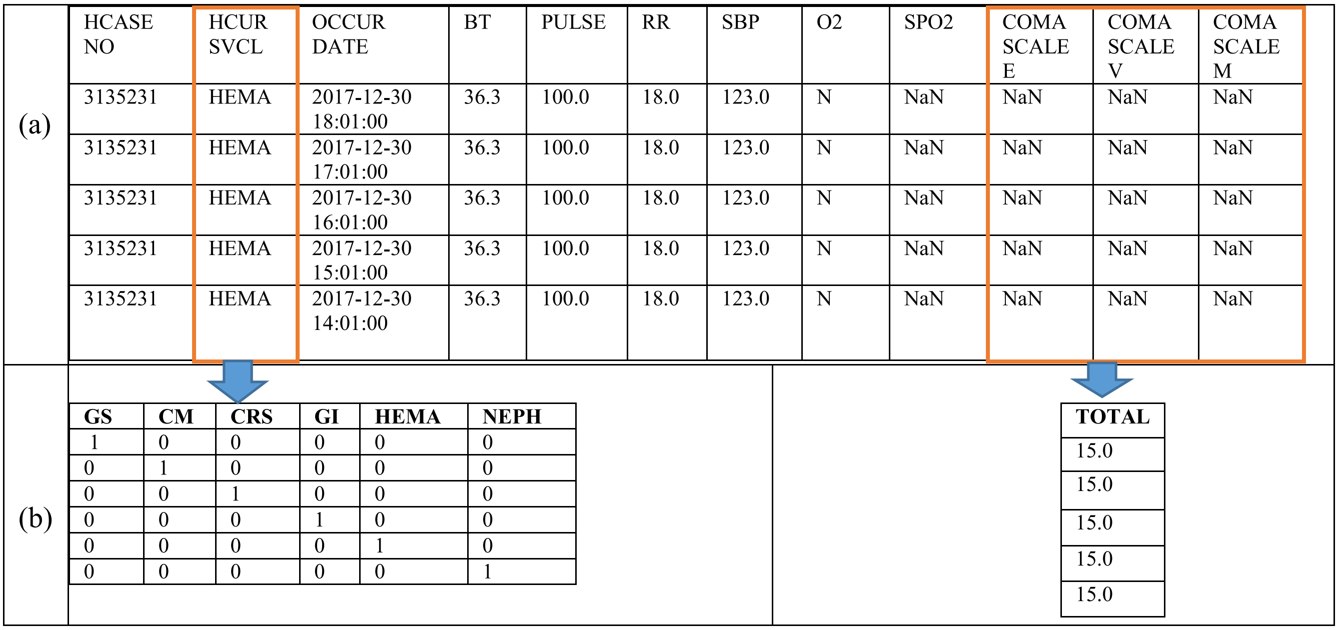
$$s^{(t)} = g^{(t)} \odot i^{(t)} + f^{(t)} \odot s^{(t-1)} \tag{9}$$

Long Short Term Memory (LSTM) network used here is to replace hidden layer with memory cell c as an intermediator storage, containing a node with weight one of recurrent edge having self-loop. In equation 9, the input node g_c at current time step input layer $x^{(t)}$ and $h^{(t-1)}$ is run through weighted input tanh activation function. The input gate i_c value uses value of input nodes to be multiplied. The internal state s_c node is with each memory cell with linear activation is a unit weight recurrent edge as self-loop also known as constant error carousel. The forget gate f_c is used to clear the internal state contents for networks continuous running. Hence, the equation 16 represents forward pass internal state calculation. The internal state s_c produced by memory cell value v_c is multiplied with value of o_c as the output gate, where tanh activation function is used to run by internal state for allotting to each cell similar dynamic range as that of hidden unit of tanh.

$$\begin{aligned} g^{(t)} &= \phi(W^{gx}x^{(t)} + W^{gh}h^{(t-1)} + b_g) \\ i^{(t)} &= \sigma(W^{ix}x^{(t)} + W^{ih}h^{(t-1)} + b_i) \\ f^{(t)} &= \sigma(W^{fx}x^{(t)} + W^{fh}h^{(t-1)} + b_f) \\ o^{(t)} &= \sigma(W^{ox}x^{(t)} + W^{oh}h^{(t-1)} + b_o) \\ h^{(t)} &= \phi(s^{(t)}) \odot o^{(t)} \end{aligned} \tag{10}$$

The equation 9 and 10, represents LSTM network having forget gates complete algorithm. However simpler LSTM can be obtained by calculating without forget gates as $f^{(t)} = 1$ for all such t. Here the input node g uses tanh activation function ô. In case of forward pass, when to allow activations is learned by LSTM for input and output gates. Hence, activation trap can occur when there is a closing of both input and out gates, whereas error in and out are learned by the gates. Thus the use of LSTM is preferred over RNNs is due to learning of high range dependency of phenomenal ability.

The Figure 3. II. shows 2-layer stacked RNN with LSTM of 128 hidden layers having input size of 15, timestamp of 28 and output to be predicted is a single scalar with sigmoid function. As the vital signs data is captured from the patients



(a)

HCASE NO	HCUR SVCL	OCCUR DATE	BT	PULSE	RR	SBP	O2	SPO2	COMA SCALE E	COMA SCALE V	COMA SCALE M
3135231	HEMA	2017-12-30 18:01:00	36.3	100.0	18.0	123.0	N	NaN	NaN	NaN	NaN
3135231	HEMA	2017-12-30 17:01:00	36.3	100.0	18.0	123.0	N	NaN	NaN	NaN	NaN
3135231	HEMA	2017-12-30 16:01:00	36.3	100.0	18.0	123.0	N	NaN	NaN	NaN	NaN
3135231	HEMA	2017-12-30 15:01:00	36.3	100.0	18.0	123.0	N	NaN	NaN	NaN	NaN
3135231	HEMA	2017-12-30 14:01:00	36.3	100.0	18.0	123.0	N	NaN	NaN	NaN	NaN

(b)

GS	CM	CRS	GI	HEMA	NEPH
1	0	0	0	0	0
0	1	0	0	0	0
0	0	1	0	0	0
0	0	0	1	0	0
0	0	0	0	1	0
0	0	0	0	0	1

TOTAL
15.0
15.0
15.0
15.0
15.0

FIGURE 4. Data Preprocessing Method Results: (a) Missing Value (HCURSVCL and Glasgow Coma Scale) and (b) One-Hot Encoding (Category) and Imputation.

in the sequential form, the structure of Convolutional NN presented in Table 3. I. CNN Hyperparameter Configuration consists of two hyper tuned convolutional layers with different output shape and one fully connected layer. Therefore, the 1D-CNN used here consists of (3 × 3) kernel size, which reads the time series data from medical devices/sensors. The model designed as shown in Figure 3. a. Structure of Convolutional Neural Network is used for sequence data feature extraction and to map its internal features. Hence, for deriving the features from fixed length segments in the dataset, 1D-CNN [32] is found to be effective irrespective of the feature location within the segment.

E. EARLY WARNING SYSTEM (EWS)

All the data preprocessed and after training the machine leaning/deep learning algorithms is then used to predict an outcome by using EWS [2]. The EWS is used to predict the patients' health conditions under risk. These predictions are then used to start a new treatment and improve the health risk that could avoid facing a critical situation. The detail results are discussed in next Section IV. Results and Discussion.

IV. RESULTS AND DISCUSSION

A. DATASET

In this section, we would present all of the experiments that we have conducted successfully as a supporting result for methodology section of our proposed paper. All of the experiments performed, within this paper are done using the following system configuration as shown in table 4:

TABLE 4. System configuration.

Computing Environment:	Workstation
Processor:	Intel Core i7 – 8700K
Memory:	32 GB
Operating System:	Ubuntu 18.04 LTS
GPU/Graphics Card:	TitanV – CUDA 9

B. DATA PREPROCESSING

In this section first, we will be discussing the results generated from the data preprocessing techniques used within this paper as defined in the methodology section. These techniques are quite important to be performed before the machine and deep learning operations take place for data classification, regression and ultimately prediction. In the second part, we will see the results produced by learning algorithms for classification by comparing their effects, solving imbalance problems and ROC curve.

As shown in Figure 4.a. missing value for Glasgow Coma Scale (GCS) is used to evaluate the COMASCALE_E, COMASCALE_V and COMASCALE_M. The E, V and M variables corresponds to eye opening, verbal response and motor response respectively. These parameters are basic health evaluations for any patient's current condition. The range of wellness is given to eye, verbal and motor responses from 1-4, 1-5 and 1-6 respectively, totaling to 15, it is considered to be normal when the patient's data for such cases is found to be missing i.e. 15. Therefore, in case of GCS, missing values are treated to be normal, as in case of most patient's and hence is assigned to be 15 for all the missing records as imputation. In Figure 4.b., the category value is replaced with a binary class matrix by one-hot encoding. As we can notice, columns GS, CM, CRS, GI, HEMA and NEPH are an empty matrix, where a binary matrix is allotted i.e. value 1 are in the top left corner diagonal because category HCURSVCL is having a common term HEMA. To represent balance in the category values, binarization of the matrix is done, which is made suitable for input to the machine learning algorithms. This input to machine learning algorithms is considered to be important as the input is strongly known for affecting the output.

Note: HCASE No. is not a feature used for training/testing.

C. EVALUATION METRIC

The precision, recall and f1-score are used to evaluate the performance of different proposed models as shown in equation 11, 12 and 13 respectively, which is calculated as below:

$$\text{Precision} = \frac{T_P}{T_P + F_P} \tag{11}$$

$$\text{Recall} = \frac{T_P}{T_P + F_N} \tag{12}$$

$$\text{F1} = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{13}$$

where T_P(true positives) indicates the outcome of the patient is adverse and the model predicted correctly. True negative for outcome as healthy. False positives(F_P) indicate the model predicted the outcome as healthy while the actual value is not. False negatives(F_N) indicate predicted value is negative while actual outcome is healthy.

Accuracy depends on their values present within the respective blocks, which is shown in equation 14:

$$\text{Accuracy} = \frac{T_P + T_N}{T_P + F_N + F_P + T_N} \tag{14}$$

where is calculated by true positive added with true negative(T_N) divided by total values to determine how correct the classifier is evaluated. The loss can be simply calculated using 1 minus accuracy.

D. EVALUATION OF DIFFERENT METHODS

In the next subsection, we present the evaluation of machine and deep learning algorithms with the hospital dataset. The hospital dataset used here is divided into training and testing datasets from 2007-2015 and 2016-2017 respectively. The experiment result of using different algorithms are shown in Table 5.

TABLE 5. Confusion matrix result using different methods.

Algorithm	*Tp*	*Tn*	*Fp*	*Fn*
(a)RF	1389	31365	305	538
(b)XGBoost	1504	31122	539	423
(c)CNN+CB+Per hour	1788	31323	338	139
(d)RNN+CB+Per hour	1781	31348	313	146

[Red color: Highest Value]

We have used two machine learning models in the Table 5. (a) Random Forest and (b) XGBoost with considering a previous record from last 3 hours. Whereas, deep learning models as CNN takes 28 records due to class imbalance problem. In Table 5.c. a convolutional neural network model has input data of stack series records and parameters as learning rate of 0.01, batch size of 128, The CNN used here is with two conv2d layers with no pooling and convolution is used to better fit the results as by using heuristics. Successively, the next version was CNN with class balance and input data as stack series records with class balance and parameters of two conv2d layer. As in our case, the CNN with the input class balance and align by per hour, the CNN model used here is again the same two conv2d layers. In Table 5.d., we have RNN with input data of align by per hour with class balance and structure of two stacked LSTM with 128 units each of 15 features and 28 timestamps.

The Figure 5. presents the performance statistics for the adverse event prediction. Here, the x-axis presents the output value as the probability score produced at the last level of CNN with CB and using per hour data by the soft-max and y-axis as the number of patients treated in the hospital. In Figure 5.a. selecting the highest cut-off range attempts to save more patients as recall score achieved is the highest (0.96) but the precision suffers (0.71). Therefore, in Figure 5.b. while tuning by optimal cut-off value range, it is observed that better recall score (0.95) is achieved. The cut point chosen is to leverage the accuracy and balance positive range for saving the patients with simultaneously managing the false alarm. The importance of cut point is high, as the shift in direction towards left will save the patient lives up to some considerable limit else there is lack in precision.

Whereas, in Figure 5.c. setting a too high cut-off range value i.e. shift towards the right, will lead to many false alarm within the system and lack in the recall score (0.90). For Figure 5, the values are detailed as in Table 6. experiment results with different cut-off lengths. When we want to deploy the model into the real-life clinical practice, we have to allow some false negative cases appear to decrease the barriers of the physicians because of alarm fatigue. Table 6 provides the adequate information to trade-off the gain and loss of the implement-tation. Also, it was observed from the Figure 5.a. and 5.c. by the doctors and hospital staff that setting a imprecise output range generated many false alarms that lead to chaos and cause misunderstandings with the deep learning system. Therefore, the training provided by the AEP-DLA research team to the group of doctors and nurses was crucial to analyze, learn, tune the system to optimal cut-off range, interpret the results and make further decision for the treatment process.

In Table 7, the experiment results of various algorithms are presented. In Table 7.a. we present the precision, recall and AUC generated by random forest algorithm. The parameters considered for this model are n estimates as 300, max depth of 27, min sample split of 30, max features of 3, oob score set to true and random state of 10. Table 7.b. XGBoost classifier is presented with parameters learning rate as 0.1, n estimators of 240, max depth of 4, min child weight of 5, random state of 10, subsample as 0.9, column sample by tree as 0.6, gamma

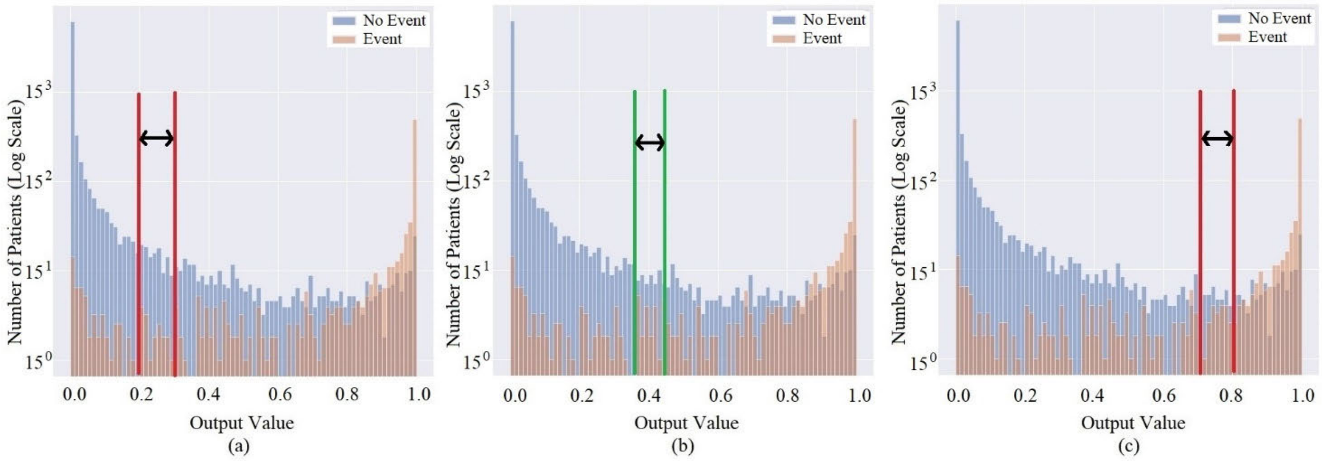


FIGURE 5. Performance statistics for (a) Attempt to save more patients in the adverse event, (b) Optimal cut-off range and (c) High precision range.

TABLE 6. Experiment result using different cut-off lengths.

	Cutoff	Outcome	Precision	Recall	F1-score	True Positive	True Negative	False Positive	False Negative
(a)	0.20	0	0.99	0.98	0.99	30910	1855	751	72
		1	0.71	0.96	0.84				
	0.30	0	0.99	0.98	0.99	31103	1835	558	92
		1	0.77	0.95	0.86				
(b)	0.35	0	0.99	0.98	0.99	31175	1827	486	100
		1	0.79	0.95	0.87				
	0.45	0	0.99	0.99	0.99	31275	1799	386	128
		1	0.82	0.93	0.88				
(c)	0.70	0	0.99	0.99	0.99	31435	1753	226	174
		1	0.88	0.90	0.90				
	0.79	0	0.99	0.99	0.99	31484	1727	177	200
		1	0.91	0.90	0.90				

TABLE 7. Experiment result using different algorithms.

Prediction Time / Reference	3 hours			12 hours			24 hours			36 hours			48 hours		
	P	R	A	P	R	A	P	R	A	P	R	A	P	R	A
Random Forest	.820	.721	.960	.781	.681	.956	.719	.573	.938	.744	.581	.933	.662	.480	.911
XGBoost	.736	.781	.966	.687	.775	.961	.611	.663	.946	.633	.673	.939	.492	.616	.916
RNN+CB+Per Hour	.850	.924	.990	.849	.904	.985	.782	.884	.980	.852	.873	.970	.737	.827	.965
AEP-DLA (CNN+CB+Per Hour)	.841	.928	.995	.890	.905	.989	.780	.868	.978	.748	.892	.979	.744	.811	.961

Symbols [P - Precision, R - Recall and A – AUC] and Color [Red: Best Score].

as 2, regularized alpha as 0.1 and evaluation metric as AUC. It is shown that for the last 3 hours, CNN with CB (class balance) and per hour pre-processing has low precision, and while RNN with CB and per hour pre-processing has better precision but the recall and AUC is low. Therefore, we chose the method which has the best recall and AUC score for the last 3 hours' adverse event prediction. The precision in RF is highest in the machine learning algorithm. Whereas, in case of recall, CNN + CB + Per Hour is the best. CNN + CB + Per Hour has better recall and AUC score in comparison using the per hour data provides better results comparing to the other methods.

TABLE 8. Depth empirical analysis of AEP-DLA.

Prediction Time / Reference	3 hours			12 hours			24 hours			36 hours			48 hours		
	P	R	A	P	R	A	P	R	A	P	R	A	P	R	A
[5]	.751	.543	.914	.638	.391	.876	.638	.362	.861	.547	.282	.828	.558	.262	.812
[8]	.555	.729	.925	.489	.601	.888	.478	.540	.878	.421	.492	.835	.411	.449	.831
[9]	.739	.541	.865	.675	.384	.835	.616	.338	.809	.568	.279	.777	.495	.256	.751
[22]	.771	.718	.944	.693	.741	.965	.640	.583	.924	.706	.623	.953	.572	.532	.902
RNN+CB+ Per Hour	.850	.924	.990	.849	.904	.985	.782	.884	.980	.852	.873	.970	.737	.827	.965
AEP-DLA (CNN+CB +Per Hour)	.841	.928	.995	.890	.905	.989	.780	.868	.978	.748	.892	.979	.744	.811	.961

Symbols [P - Precision, R - Recall and A – AUC] and Color [Red: Best Score].

In Table 8, explains the experiments results obtained from the AI model for the references benchmark comparison with RNN and CNN. The red color highlighting indicates the highest score achieved in that respective algorithm comparison. The AEP-DLA exceeds in the benchmark comparison for recall with accuracy and proves its worth for the implementation.

E. EVALUATION OF OUR PROPOSED METHOD

To verify the performance of our proposed method, in this section we compared the model with other methods. The EDI [19] uses Naïve bayes to calculate continuous risk scores from their vital signs data. NEWS [5] is a popular scoring system and standard adopted worldwide for patients with severity of acute-illness. Using logistic regression to predict early clinical deterioration after ICU transfer while MEWS [20] is an improved version of NEWS, also using logistic regression as their proposed method to predict the injury severity, ICU resource usage, air transport and mortality. Kwon *et al.* [29], [30] proposed a 3-layer LSTM for cardiac arrest during hospitalization. The risk stratification tool is one of the analysis method for medical data [31]. They have used 8 hours' data for their model, while the other three methods mentioned above uses 1 hour data to predict the risk.

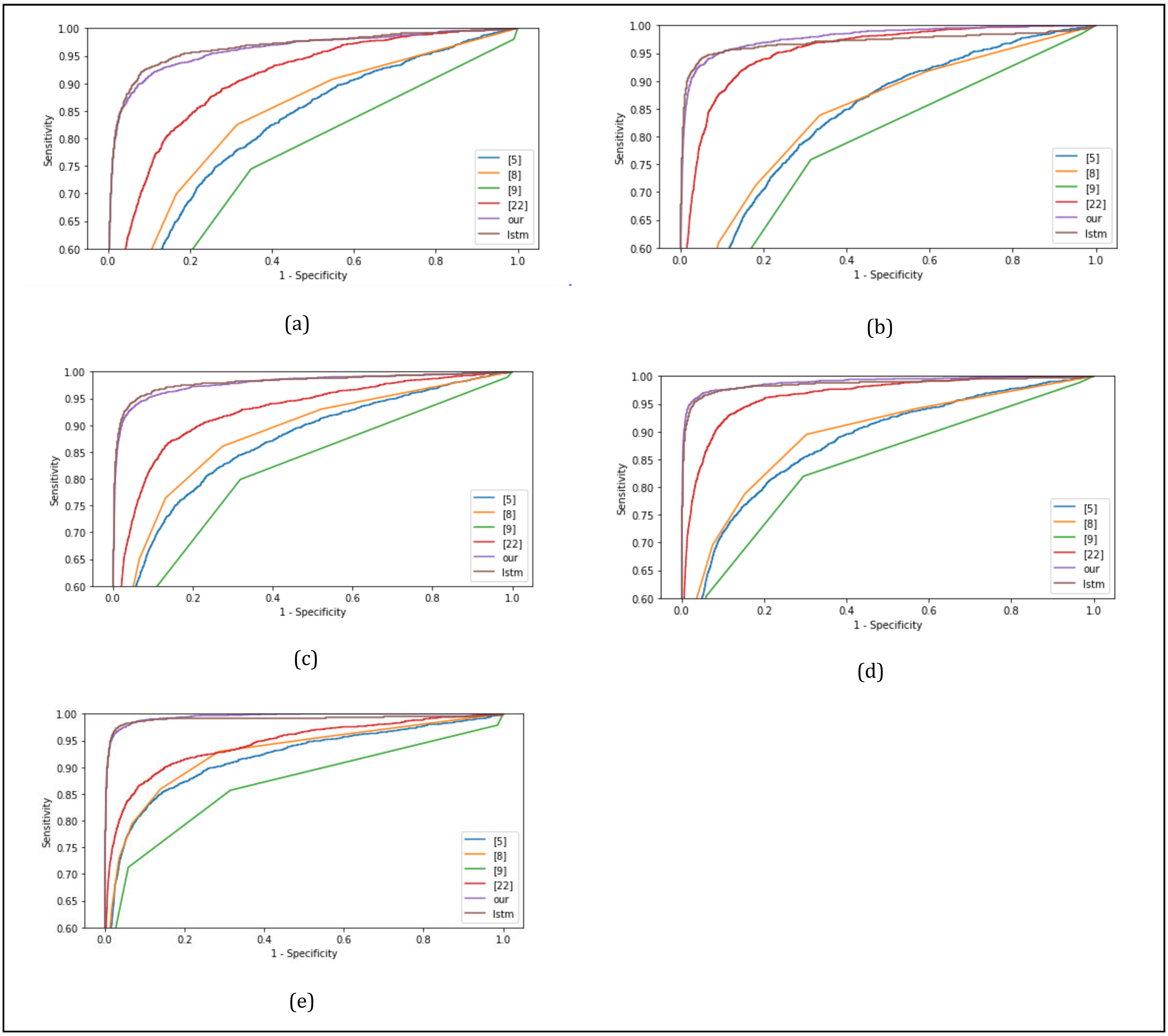
Our model uses 28 hours' data, which is in time series format collecting the vital sign features of the patient continuously from the time of admittance to the hospital, instead of just preliminary single record comparison from the registration report with adjacent research studies. We experiment our data with the methods mentioned above to predict the risk 3 hours later. In Table 7. The results show the Convolutional Neural Network of AEP-DLA performs better in the last 3 hours for recall and AUC, than the other models. It is shown that our method has the greatest performance on the hospital data. Both precision and recall are better compared to the other methods. Therefore, AEP-DLA uses convolutional neural network as the preferred choice for the system model. The results available from our model for last 28 hours as the health trend before the adverse event presents the effective use of vital signs data collected continuously from the patient's admittance to provide early diagnosis and treatment.

V. CONCLUSION

Adverse event prediction is considered to be crucial for saving the patient's life and improving health conditions. The physicians usually check the last data and decide what to do next every day. Therefore, we took the last 28 hours vital signs into the AEP-DLA model. Hence, once the patient is admitted to the hospital after the first 28 hours, the algorithm starts processing vital signs and later to predict the results. Thus, the alert raised by the system within last 28 hours will help to seek the doctor's attention for the emergency situation and to carefully handle the critical case by treatment. To provide world-class facilities and overcome the various health risk, the proposed AEP-DLA provides various data pre-processing capabilities, by handling of missing values, process on a single record and multiple records using machine learning and deep learning classifiers respectively to predict better outcomes. The key to achieve better performance using deep learning was to apply good data pre-processing strategy i.e. stack series record method and align by per hour record method, for appropriate availability of the input as vital signs patient records for the prediction. Henceforth, admitted patient's health records are analyzed, disease severity can be determined and the adverse event can be predicted before a substantial amount of time. The method using CNN + CB + Per Hour pre-processing has proved to have best result in benchmark comparison of 92.8% recall and 99.5% AUC score, as the data was first being sorted in hours, some pre-processing and balancing classes were also performed. Various experiments are conducted and proved that not only the method includes most of the features, it also provides better performance prediction on the hospital data. In future, we have planned to apply explainable AI to improve this model and provide detail design insights.

TABLE 9. Anova significance test on the dataset.

Source of Variation	*SS*	*df*	*MS*	*F*	*P-value*	*F crit*
Between Groups	7.81E+12	27	2.89E+11	99621.24	0	1.48568
Within Groups	9.02E+12	3107564	2902850			
Total	1.68E+13	3107591				



(a)

(b)

(c)

(d)

(e)

FIGURE 6. ROC curve generated for (a) 48 hrs, (b) 36 hrs, (c) 24 hrs, (d) 12 hrs and (e) 3 hrs for the comparison with different models with reference [5], [8], [9], [22], ours (AEP-DLA) and LSTM/RNN.

APPENDIX

The Table 9. shows Analysis of Variance (anova) significance test for the dataset and have found the means spread across the different features/columns. As F > F crit., we reject the null hypothesis. Therefore, the spread across the different features are quite significant. The Figure 6. presents ROC curves for all the models as referenced from the Table 8. for depth empirical analysis indicating AEP-DLA performs as best in comparison. ROC is used to evaluate performance of a binary classifier, whereas AUC curve score is the single value performance summary.

REFERENCES

[1] D. W. Bates and H. Singh, ''Two decades SinceTo err is human: An assessment of progress and emerging priorities in patient safety,'' *Health Affairs*, vol. 37, no. 11, pp. 1736–1743, Nov. 2018.

[2] B. A. Goldstein, A. M. Navar, M. J. Pencina, and J. Ioannidis, ''Opportunities and challenges in developing risk prediction models with electronic health records data: A systematic review,'' *J. Amer. Med. Inform. Assoc.*, vol. 24, no. 1, pp. 198–208, Jan. 2017.

[3] G. J. Escobar, V. X. Liu, A. Schuler, B. Lawson, J. D. Greene, and P. Kipnis, ''Automated identification of adults at risk for in-hospital clinical deterioration,'' *New England J. Med.*, vol. 383, no. 20, pp. 1951–1960, Nov. 2020.

[4] S. Gerry, T. Bonnici, J. Birks, S. Kirtley, P. S. Virdee, P. J. Watkinson, and G. S. Collins, ''Early warning scores for detecting deterioration in adult hospital patients: Systematic review and critical appraisal of methodology,'' *BMJ*, vol. 369, p. m1501, May 2020.

[5] R. C. O. Physicians, ''National early warning score (NEWS): Standardizing the assessment of acute-illness severity in the NHS. London,'' Working Party, RCP, London, U.K., Tech. Rep., 2012.

[6] R. C. O. Physicians, ''National early warning score (NEWS) 2: Standardizing the assessment of acute-illness severity in the NHS,'' Working Party, RCP, London, U.K., Tech. Rep., 2017.

[7] S. L. Hyland, M. Faltys, M. Hüser, X. Lyu, T. Gumbsch, C. Esteban, C. Bock, M. Horn, M. Moor, B. Rieck, M. Zimmermann, D. Bodenham, K. Borgwardt, G. Rätsch, and T. M. Merz, ''Early prediction of circulatory failure in the intensive care unit using machine learning,'' *Nature Med.*, vol. 26, no. 3, pp. 364–373, Mar. 2020.

[8] H. Mohamadlou, S. Panchavati, J. Calvert, A. Lynn-Palevsky, S. Le, A. Allen, E. Pellegrini, A. Green-Saxena, C. Barton, G. Fletcher, L. Shieh, P. B. Stark, U. Chettipally, D. Shimabukuro, M. Feldman, and R. Das, ''Multicenter validation of a machine-learning algorithm for 48-h all-cause mortality prediction,'' *Health Informat. J.*, vol. 26, no. 3, pp. 1912–1925, Sep. 2020.

[9] Y. D. Chiu, S. S. Villar, J. W. Brand, M. V. Patteril, D. J. Morrice, J. Clayton, and J. H. Mackay, ''Logistic early warning scores to predict death, cardiac arrest or unplanned intensive care unit re-admission after cardiac surgery,'' *Anaesthesia*, vol. 75, no. 2, pp. 162–170, Feb. 2020.

[10] M. E. H. Chowdhury, T. Rahman, A. Khandakar, S. Al-Madeed, S. M. Zughaier, S. A. R. Doi, H. Hassen, and M. T. Islam, ''An early warning tool for predicting mortality risk of COVID-19 patients using machine learning,'' Jul. 2020, *arXiv:2007.15559*. [Online]. Available: http://arxiv.org/abs/2007.15559

[11] A. Kia, P. Timsina, H. N. Joshi, E. Klang, R. R. Gupta, R. M. Freeman, D. L. Reich, M. S. Tomlinson, J. T. Dudley, R. Kohli-Seth, M. Mazumdar, and M. A. Levin, ''MEWS++: Enhancing the prediction of clinical deterioration in admitted patients through a machine learning model,'' *J. Clin. Med.*, vol. 9, no. 2, p. 343, Jan. 2020.

[12] K.-J. Cho, O. Kwon, J.-M. Kwon, Y. Lee, H. Park, K.-H. Jeon, K.-H. Kim, J. Park, and B.-H. Oh, ''Detecting patient deterioration using artificial intelligence in a rapid response system,'' *Crit. Care Med.*, vol. 48, no. 4, pp. e285–e289, Apr. 2020.

[13] S. M. Lauritsen, M. Kristensen, M. V. Olsen, M. S. Larsen, K. M. Lauritsen, M. J. Jørgensen, J. Lange, and B. Thiesson, ''Explainable artificial intelligence model to predict acute critical illness from electronic health records,'' *Nature Commun.*, vol. 11, no. 1, pp. 1–11, Dec. 2020.

[14] A. Abdulaal, A. Patel, E. Charani, S. Denny, N. Mughal, and L. Moore, ''Prognostic modeling of COVID-19 using artificial intelligence in the united kingdom: Model development and validation,'' *J. Med. Internet Res.*, vol. 22, no. 8, Aug. 2020, Art. no. e20259.

[15] D.-Y. Kang, K.-J. Cho, O. Kwon, J.-M. Kwon, K.-H. Jeon, H. Park, Y. Lee, J. Park, and B.-H. Oh, ''Artificial intelligence algorithm to predict the need for critical care in prehospital emergency medical services,'' *Scandin. J. Trauma, Resuscitation Emergency Med.*, vol. 28, no. 1, pp. 1–8, Dec. 2020.

[16] H.-K. Chang, C.-T. Wu, J.-H. Liu, and J.-S.-R. Jang, ''Using machine learning algorithms in medication for cardiac arrest early warning system construction and forecasting,'' in *Proc. Conf. Technol. Appl. Artif. Intell. (TAAI)*, Nov. 2018, pp. 1–4.

[17] M. E. B. Smith, J. C. Chiovaro, M. O'Neil, D. Kansagara, A. R. Quiñones, and M. Freeman, ''Early warning system scores for clinical deterioration in hospitalized patients: A systematic review,'' *Ann. Amer. Thoracic Soc.*, vol. 11, pp. 65–1454, Nov. 2014.

[18] L. S. van Galen, C. C. Dijkstra, J. Ludikhuize, M. H. H. Kramer, and P. W. B. Nanayakkara, ''A protocolised once a day modified early warning score (MEWS) measurement is an appropriate screening tool for major adverse events in a general hospital population,'' *PLoS ONE*, vol. 11, no. 8, Aug. 2016, Art. no. e0160811.

[19] E. Ghosh, L. Eshelman, L. Yang, E. Carlson, and B. Lord, ''Early deterioration indicator: Data-driven approach to detecting deterioration in general ward,'' *Resuscitation*, vol. 122, pp. 99–105, Jan. 2018.

[20] M. M. Churpek, A. Snyder, X. Han, S. Sokol, N. Pettit, M. D. Howell, and D. P. Edelson, ''Quick sepsis-related organ failure assessment, systemic inflammatory response syndrome, and early warning scores for detecting clinical deterioration in infected patients outside the intensive care unit,'' *Amer. J. Respiratory Crit. Care Med.*, vol. 195, no. 7, pp. 906–911, Apr. 2017.

[21] M. Green, H. Lander, A. Snyder, P. Hudson, M. Churpek, and D. Edelson, ''Comparison of the between the flags calling criteria to the MEWS, NEWS and the electronic cardiac arrest risk triage (eCART) score for the identification of deteriorating ward patients,'' *Resuscitation*, vol. 123, pp. 86–91, Feb. 2018.

[22] T. Desautels, J. Calvert, J. Hoffman, M. Jay, Y. Kerem, L. Shieh, D. Shimabukuro, U. Chettipally, M. D. Feldman, C. Barton, D. J. Wales, and R. Das, ''Prediction of sepsis in the intensive care unit with minimal electronic health record data: A machine learning approach,'' *JMIR Med. Informat.*, vol. 4, no. 3, p. e28, Sep. 2016.

[23] B. Wellner, J. Grand, E. Canzone, M. Coarr, P. W. Brady, J. Simmons, E. Kirkendall, N. Dean, M. Kleinman, and P. Sylvester, ''Predicting unplanned transfers to the intensive care unit: A machine learning approach leveraging diverse clinical elements,'' *JMIR Med. Informat.*, vol. 5, no. 4, p. e45, Nov. 2017, doi: 10.2196/medinform.8680.

[24] M. M. Churpek, T. C. Yuen, C. Winslow, D. O. Meltzer, M. W. Kattan, and D. P. Edelson, ''Multicenter comparison of machine learning methods and conventional regression for predicting clinical deterioration on the wards,'' *Crit. Care Med.*, vol. 44, no. 2, pp. 368–374, Feb. 2016, doi: 10.1097/CCM.0000000000001571.

[25] S. Saadat, A. Aziz, H. Ahmad, H. Imtiaz, Z. S. Sohail, A. Kazmi, S. Aslam, N. Naqvi, and S. Saadat, ''Predicting quality of life changes in hemodialysis patients using machine learning: Generation of an early warning system,'' *Cureus*, vol. 9, no. 9, p. e1713, Sep. 2017, doi: 10.7759/cureus.1713.

[26] C. Bishop, *Neural Networks for Pattern Recognition*. Oxford, U.K.: Clarendon Press, 1995.

[27] S. Ioffe and C. Szegedy, ''Batch normalization: Accelerating deep network training by reducing internal covariate shift,'' in *Proc. Int. Conf. Mach. Learn.*, Paris, France, vol. 37, Jul. 2015, pp. 448–456.

[28] Z. C. Lipton, J. Berkowitz, and C. Elkan, ''A critical review of recurrent neural networks for sequence learning,'' Oct. 2015, *arXiv:1506.00019*. [Online]. Available: http://arxiv.org/abs/1506.00019

[29] J.-M. Kwon, Y. Lee, Y. Lee, S. Lee, H. Park, and J. Park, ''Validation of deep-learning-based triage and acuity score using a large national dataset,'' *PLoS ONE*, vol. 13, no. 10, Oct. 2018, Art. no. e0205836, doi: 10.1371/journal.pone.0205836.

[30] J. Kwon, Y. Lee, Y. Lee, S. Lee, and J. Park, ''An algorithm based on deep learning for predicting in-hospital cardiac arrest,'' *J. Amer. Heart Assoc.*, vol. 7, no. 13, Jul. 2018, Art. no. 008678, doi: 10.1161/JAHA.118.008678.

[31] M. M. Churpek, T. C. Yuen, C. Winslow, A. A. Robicsek, D. O. Meltzer, R. D. Gibbons, and D. P. Edelson, ''Multicenter development and validation of a risk stratification tool for ward patients,'' *Amer. J. Respiratory Critical Care Med.*, vol. 190, pp. 55–649, Sep. 2014.

[32] W. Tang, G. Long, L. Liu, T. Zhou, J. Jiang, and M. Blumenstein, ''Rethinking 1D-CNN for time series classification: A stronger baseline,'' Feb. 2020, *arXiv:2002.10061*. [Online]. Available: http://arxiv.org/abs/2002.10061

CHIEH-LIANG WU received the M.D. and Ph.D. degrees. He is currently working with the Department of Critical Care Medicine, Taichung Veterans General Hospital, Taichung, Taiwan. He has been with the Department of Automatic Control Engineering, Feng Chia University, Taichung, and the Department of Industrial Engineering and Enterprise Information, Tunghai University, Taichung.

MING-JU WU received the M.D. and Ph.D. degrees. He worked with the Rong Hsing Research Center for Translational Medicine, College of Life Science, Institute of Biomedical Science, National Chung Hsing University, Taichung, Taiwan. He is currently working with the Division of Nephrology, Department of Internal Medicine, Taichung Veterans General Hospital, Taiwan. He is also a Professor with the School of Medicine, Chung Shan Medical University, Taichung.

LUN-CHI CHEN received the M.S. degree in computer science from Tunghai University, Taiwan, in 2005, and the Ph.D. degree from National Chung-Hsing University, in 2015. Since 2006, he has been with the National Center for High-Performance Computing, Taiwan, where he was an Associate Researcher, from 2007 to 2018. He has been an Assistant Professor with the College of Engineering, Tunghai University, since 2018. His current research interests include data analytics, deep learning, text mining, and cloud computing.

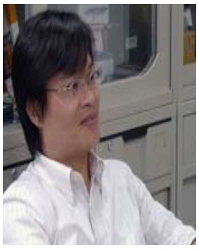
YING-CHIH LO received the M.D. and Ph.D. degrees. He was with the Harvard Medical School, Boston, MA, USA. He is currently working with the Center of Quality Management, Taichung Veterans General Hospital, Taichung, Taiwan. He has been with Department of Data Science and Big Data Analytics, Providence University, Taichung, the Division of General Internal Medicine and Primary Care, Brigham, and the Women's Hospital, Boston.

CHIEN-CHUNG HUANG received the B.S. degree. He is currently working with the Computer and Communication Center, Taichung Veterans General Hospital, Taichung, Taiwan, and the Department of Computer Science, Tunghai University, Taiwan.

HSIU-HUI YU received the M.S.N. and R.N. degrees. She is currently working with the Department of Nursing, Taichung Veterans General Hospital, Taichung, Taiwan.

MAYURESH SUNIL PARDESHI received the B.E. degree in information technology from Mumbai University, in 2010, the M.Tech. degree in computer science and engineering (CSE) from the Walchand College of Engineering, Sangli, India, in 2013, and the Ph.D. degree in electrical engineering and computer science (EECS-IGP) from National Chiao Tung University, Hsinchu, Taiwan, in 2020. He is currently a Postdoctoral Researcher with the AI Center. His current research interests include artificial intelligence and security in distributed systems. He is also serving as a Reviewer and a Sub-Reviewer for many international journals.

WIN-TSUNG LO received the M.S. and Ph.D. degrees from the University of Maryland, USA. He is currently a Professor with the Department of Computer Science, Tunghai University, Taiwan. He is also active as the Director of the Cloud Innovation School (CIS) and the AI Center. His current research interests include distributed computing and AI in healthcare, industry, and construction.



RUEY-KAI SHEU received the M.S. and Ph.D. degrees from National Chiao Tung University, in 1998 and 2001, respectively. He joined W&Jsoft Inc., as the Research and Development Leader, from 1999 to 2007. He has been an Associate Professor with the Department of Computer Science, Tunghai University, Taichung, Taiwan, since 2014. His current research interests include landing of machine learning or artificial intelligence applications, cloud systems, and data leak protection technologies.