Biomedical Signal Processing and Control 100 (2025) 106975

Contents lists available at ScienceDirect

Biomedical Signal Processing and Control

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

Temporal variational autoencoder model for in-hospital clinical emergency prediction

Trong-Nghia Nguyen [a], Soo-Hyung Kim [a,*], Bo-Gun Kho [b,*], Nhu-Tai Do [a], Ngumimi-Karen Iyortsuun [a], Guee-Sang Lee [a], Hyung-Jeong Yang [a]

[a] Department of Artificial Intelligence Convergence, Chonnam National University, Gwangju, Korea
[b] Pulmonology and Critical Care Medicine, Chonnam National University Hospital, Gwangju, Korea

ARTICLE INFO

Keywords:
Clinical deterioration
Rapid response system
Deep learning
Clinical medical signal

ABSTRACT

Early recognition of clinical deterioration plays a pivotal role in a Rapid Response System (RRS), and it has been a crucial step in reducing inpatient morbidity and mortality. Traditional Early Warning Scores (EWS) and Deep Early Warning Scores (DEWS) for identifying patients at risk are still limited because of the challenges of imbalanced multivariate temporal data. Typical issues leading to their limitations are low sensitivity, high late alarm rate, and lack of interpretability; this hinders the system's deployment in clinical settings. This study develops an EWS based on Temporal Variational Autoencoder (TVAE) and a window interval learning framework, which uses the latent space features generated from the input multivariate temporal features, to learn the temporal dependence distribution between the target labels (clinical deterioration probability). Implementing the target information in the Fully Connected Network (FCN) with a loss function assists in addressing the imbalance problem and improving the performance of the time series classification task. The proposed method is validated on an in-house clinical dataset collected from Chonnam National University Hospitals (CNUH) and a public dataset from the University of Virginia (UV). Extensive trials with diverse validation methods and study cases demonstrate that our system outperforms existing methods, showcasing remarkable performance across usual criteria for classification models, typically as the Area Under The Receiver Operating Characteristic Curve (AUROC) and the Area Under The Precision-Recall Curve (AUPRC). TVAE exhibits a notable decrease in late alarm issues across two datasets, presenting stable performance even with a limited number of data samples.

1. Introduction

In-hospital mortality and cardiac arrest are considerable burdens to public health, affecting the quality of the healthcare system and making patients incur considerable health costs. Of the patients suffering from in-hospital cardiac arrest, 80% show early signs of clinical deterioration in the 8 h before the event time [1,2]. Therefore, controlling the early clinical deterioration of patients and providing timely treatments are vital for reducing cardiac arrest complications. Researchers and doctors have taken advantage of improvements in modern scientific equipment and rapid access to Electronic Health Records (EHRs) [3] to prevent in-hospital cardiac arrest and mortality. However, the COVID-19 pandemic in the early 2020s has limited the effectiveness of healthcare systems in caring for patients with cardiac arrests. Consequently, the in-hospital mortality rate increases because of the specificity of the disease and overload of the health system, leaving critically ill patients not receiving timely treatment. In such circumstances, the flexibility and efficiency of the medical team are of paramount necessity. In this context, the importance of RRS [4,5] becomes more obvious.

RRS has three main elements: the afferent limb, the efferent limb, and administration (Fig. 1). The afferent limb, also known as the Track-And-Trigger System (TTS), detects and predicts patient deterioration [6], which can be conducted manually by a nurse or physician (monitoring and communicating with the response team). This system can be based on a risk score combined with a tracking system from the EHR. This study uses the afferent limb to develop a risk score for assessing and detecting patient deterioration and enhancing RRS performance.[1]

In the afferent limb, the risk score is one of the most common methods currently used to assess the patient's clinical status. Vital signs and laboratory tests are the common parameters for assessing this

* Corresponding authors.
E-mail addresses: shkim@jnu.ac.kr (S.-H. Kim), imdrkbg@gmail.com (B.-G. Kho).
[1] Source available at: https://github.com/nghianguyen7171/TVAE-RRS.git.

https://doi.org/10.1016/j.bspc.2024.106975
Received 21 February 2023; Received in revised form 3 May 2024; Accepted 21 September 2024
Available online 7 October 2024

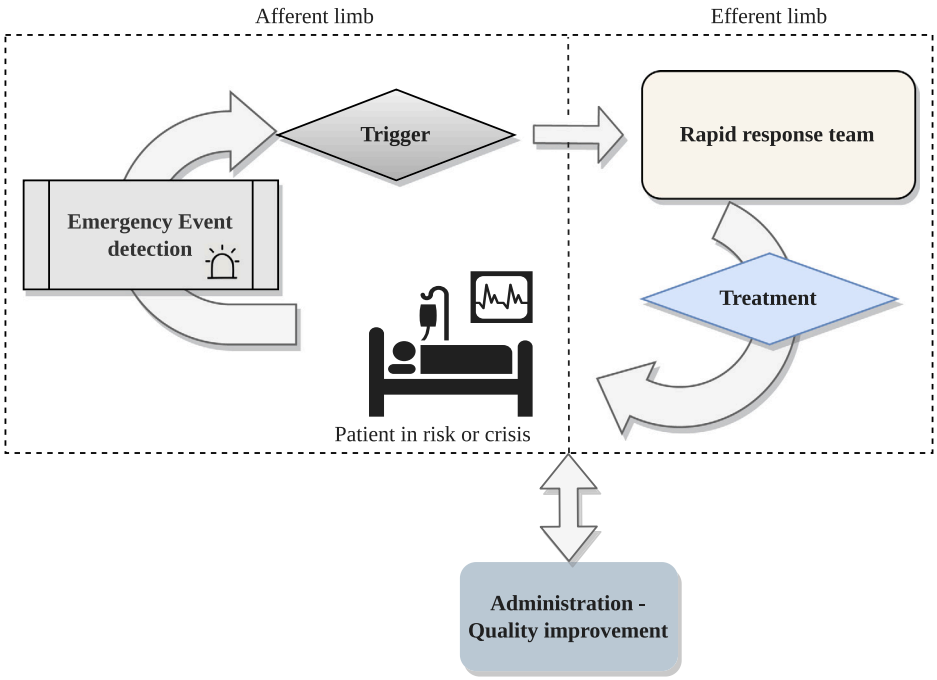


Fig. 1. An overview of RRS, includes three main components: afferent limb, efferent limb, and administration. The afferent limb involves the mechanisms for recognizing and escalating concerns about a patient's deteriorating state. The efferent limb is responsible for addressing the identified concerns within the system. This component stresses the timely and effective actions taken to manage the patient's deteriorating health. The administration component is in charge of the general coordination and management of RRS. This aspect ensures the system's efficiency, responsiveness, and ongoing refinement.

score, measuring, and storing them on the EHR [7]. The TTS [8] has been integrated for early detection of patients' clinical deterioration or cardiac arrest based on their risk score [9,10]. The EWS, as a Modified Early Warning Score (MEWS) [11], is the most common approach among TTSs [12]. Deep learning (DL) techniques have been widely applied in the medical field, especially in medical image processing [13]. For temporal data, such as patients' clinical symptom information, DL approaches are problematic because of their spontaneity and irregularity. This has been a significant challenge for RRS because the input of data into the system, measured during or at post-ICU admission, has an unstable quality and is often influenced by external factors [14]. Until 2018, DEWS has received significant attention. DEWS is a DL-based method for evaluating and predicting patient in-hospital mortality and cardiac arrest rate through event probability. Motivated by DEWS performances, this study develops a novel DL-based approach for RRS. Based on the inherent principles of DEWS, we build a model based on the patient's abnormal prognosis probability. Our focus is to overcome the inherent problems of medical data and time series data, such as naturalness and data imbalance. This study provides the following contributions:

- This research marks the inaugural development of a DL application system designed to enhance the track-and-trigger stage within the RRS.
- The proposed approach increases the performance and minimizes the occurrence of late alarms in the prediction system, hence providing crucial support to the medical team in their decision-making processes. The model's LSTM encoder-to-decoder structure increases the system's interpretability through temporal extraction, optimizing the information of time series input data.
- The proposed system introduces a novel contribution by leveraging the Variational Autoencoder (VAE) structure to extract robust latent representation. These features increase the classification system's effectiveness by offering a more discriminative and generalized representation for better predictions.
- In the prediction part, we implement imbalance loss along with focal loss for the FCN structure to overcome the high late alarm rate problems, which are particularly dangerous in clinical deterioration prediction tasks. The optimal multi-task learning tackles the imbalance issue of the multivariate medical dataset.
- This extensive experiment, which includes numerous cross-validation methods, case study configurations, and data size variants, thoroughly verifies the proposed methods' stability and superior performance when compared to competitive approaches.

2. Related works

Recently, Machine Learning (ML) and DL techniques have achieved significant success in clinical signal medical tasks.

For ML model, trees-based and boosting models show high efficiency in biomedical signal processing [15,16]. Researchers in [15] proposed an ML-based system to predict out-of-hospital cardiac arrest. The ML models have shown impressive results on a large nationwide database in Japan with 0.882, 0.866, and 0.877 of the AUROC for Logistic Regression (LR), Support Vector Machine (SVM), and Random Forest (RF), respectively. The works in [16] aims to develop a mortality prediction model for Acute Kidney Injury (AKI) patients in ICU using the Extreme Gradient Boosting Machine (XGBM) model and compare its performance with LR, SVM, and RF models. The XGBM model demonstrated superior predictive performance, with the highest AUROC, F1 score, and accuracy among the four models.

DL models designed to address this problem can be categorized into two primary structures: Non-Recurrent Neural Network (Non-RNN) [17–19] models and RNN models [20–22].

For non-RNN structures, study [17] addresses the limitations of traditional bio-statistical scores in ICU by proposing a Deep Convolutional Network (DCNN) architecture for mortality prediction using the Medical Information Mart for Intensive Care III (MIMIC-III) dataset. The model achieves competitive performance with state-of-the-art deep learning mortality models on MIMIC-III data while maintaining interpretability through visual explanations based on coalitional game theory concepts. The authors in [18] investigate the use of a machine learning algorithm, employing Fully Convolutional Neural Networks

(FCNN) and EHR data, to predict AKI up to 48 h in advance, addressing the challenging nature of early AKI detection.

For RNN structures, Kwon et al. [20] developed a DEWS-based RNNs and Long Short-Term Memory (LSTM) unit for the temporal classification task, and they were the first to predict cardiac arrest using DL. Their model detected more than a 50% decline in the in-hospital clinical 14 h before the event. Their model outperformed the traditional EWS with 24.3% higher sensitivity and a reduced 41.6% false alarm rate. Shamout et al. [21] proposed a Deep Interpretable End-to-End system by inputting the time-series data using the Gaussian process regression (GPR) [23] to predict the patient's clinical event probability. The architect of their model was an encoder with Bidirectional-LSTM (BiLSTM) units and attention block [24], which computed the context vectors of the mean and variance of input vital signs. They obtained impressive results (AUROC: 0.880) with an unsurpassed precision DEWS compared with the NEWS [25] (AUROC: 0.866), which was currently implemented in the clinical setting. Another study [22] introduced an explainable AI-EWS system based on a Temporal Convolution Network (TCN) [26] for early acute critical illness prediction and patient data explanation. Their research offers promising future opportunities. The above research offered a promising future opportunity for using DL for acute clinical care and TTS application.

However, the DEWS models using in-house medical datasets have some inherent issues when using current DL systems. Using an imbalanced dataset, the difference in the number of sample distributions between the majority class and minority class leads DEWS to estimate the minority class inaccurately, known as the high false/late alarm rates. Moreover, the complexity and high rate of data imbalance cause the DEWS to be incapable of understanding and interpreting minority classes when using natural temporal data. A low DEWS score (ranging from 0 to 100) is the consequence of this problem. The difference in this index between the two classes (poor sensitivity) causes doctors or RRS administrators to face challenges during the diagnostic process, which consequently affects the final system output. Besides, various external factors can influence the number and quality of the data throughout the RRS monitoring procedure [27,28]. This presents potential biases and limits due to limited sample numbers, such as uncertain and biased performance estimations [29,30].

Thus, we developed an innovative end-to-end DL-based framework for clinical emergency prediction leveraged using TVAE, time-series interval sliding window, and class-imbalanced multi-task learning for in-hospital clinical status prediction. Our approach uses the generative model to explore the data distribution from the original discrete clinical information. We integrated the LSTM into VAE to represent the distribution of clinical data efficiently in a temporal context. LSTM is capable of discovering local-dependent properties, while VAE discovers global properties in the data distribution, which helps our model overcome the late alarm problem. Additionally, our model separates the probability amplitudes of non-event and event classes, represented by the DEWS score. Moreover, we proposed a multi-task learning framework using a reconstruction loss to learn the class distribution of latent representation and clinical loss for the clinical deterioration prediction task. This multi-task learning framework also determines the probability of minority class (event status) and an imbalance loss to overcome the calculation imbalance in the dataset. Using the CNUH data, the experimental results showed that our model outperformed the existing models. We also used a public dataset (UV) with more extensive data to validate and obtain the best results.

3. Materials and methods

3.1. Dataset

The protocol of this study was approved by the Independent Institutional Review Board (IRB) of Chonnam National University Hospitals. Our study has received considerable attention and support from Chonnam National University Hwasun Hospital to improve the quality of emergency care. The primary database used for the experiment is licensed patient data provided by Chonnam National University Hwasun Hospital's doctors. The characteristic statistic of the experimental datasets is shown in Table 1.

3.1.1. Chonnam National University Hospital dataset (CNUH)

The in-house dataset, CNUH, contained the clinical variables obtained from 2615 patients of two hospitals of Chonnam National University. The data collection process is in two phases. The first set was assembled from the inpatient clinical dataset at Hwasun Chonnam National University Hospital from March 2017 to February 2019. From January 2019 to April 2019, the second set was collected from the inpatient clinical records of Hakdong Chonnam National University Hospital. The main variables are time-series data (vital signs and laboratory tests), static information (gender, age), and label/clinical status (non-event, event). Their mean age was 65.66 (95% CI 65.14 – 66.17). Females were 63%, and males were 36.8%.

The dataset's clinical variables included patients' vital signs and laboratory test results measured every hour for each patient. The vital sign features included the patient's five main indexes: body temperature, systolic blood pressure, heart rate, respiration rate, and oxygen saturation. However, the laboratory test contained indexes collected from a sample of the patient's blood, including alanine transaminase, blood urea nitrogen, aspartate aminotransferase, the white blood cell count, C-reactive protein, albumin, lactate, total protein, platelet, hemoglobin (Hgb), Alkaline phosphatase, total calcium, total bilirubin, Creatinine, Glucose, Sodium, Chloride, Potassium.

Two groups of patients are sampled: Event and non-event. The event (abnormal) group was patients with cardiac arrest and in-house airway intubation admitted into the internal medicine department (indigestion medicine, circulatory internal medicine, respiratory medicine, kidney internal medicine, endocrine internal medicine, blood internal medicine, oncology, infectious medicine, and allergic medicine). The non-event (normal) group included patients who did not undergo cardiopulmonary resuscitation and airway intubation during hospitalization. The imbalance of this dataset was expressed by the ratio of event patients and non-event patients with only 41 event patients (1.5%) compared with 2574 (98.5%) normal patients.

3.1.2. University of Virginia dataset (UV) [31]

To increase our framework's authenticity and probability, we validated our model on Virginia University's public dataset, which was the 63-year data of 8,105 acute care patients admitted to a tertiary care academic medical center. This dataset shares some similarities with the CNUH data. It also has the patient's vital signs and laboratory test parameters. However, it is more diverse in the number of features because it has 7 vital signs and 24 laboratory test variables.

In addition, UV included 15 cardiorespiratory dynamics features measured through continuous electrocardiography (ECG) monitoring. The ECG is a measuring tool that can check a patient's heart rate and electrical activity [32]. However, we removed these features for fairness in the experiment. The mean age was 63.74 (95% CI 63.42–64.07). Instead of gender, the data provided racial group data, comprising 80.9%, 16.9%, and 2.2% for white, black, and others, respectively. The outcomes of this database were represented as event and non-event. The event class defines cases, in which the patient is transferred to the ICU or dies. From the definition of the output of this dataset, it differs from the CNUH data. The class imbalance was also an issue in this data, with only 367 event patients (4.5%) compared with 7,738 non-event patients (95.4%). Since the number of samples and the size of this dataset is beyond our resource's capacity, we only randomly select 1000 samples from normal samples. In contrast, the number of abnormal samples was consistent.

Table 1
Experimental dataset characteristics.

Dataset	Characteristic	Value
CNUH	**Patients characteristic (n = 2,615)**	
	Age (years), mean (95% CI)	65.66 (65.14 – 66.17)
	Gender (female/male), n (%)	1,651 (63.2%)/964 (36.8%)
	Vital signs (number of features)	5 features
	Laboratory test (number of features)	18 features
	Clinical status (even/ non-event), n (%)	41 (0.015)/2574 (0.98)
UV	**Patients characteristic (n = 8,105)**	
	Age (years), mean (95% CI)	63.74 (63.42 – 64.07)
	Race (white/black/other), n (%)	6,559 (80.9%)/1,374 (16.9%)/ 172 (2.2%)
	Vital signs (number of features)	7 features
	Laboratory test (number of features)	24 features
	ECG monitoring (number of features)	15 features
	Clinical status (even/ non-event), n (%)	367 (4.5%)/7,738 (95.4%)

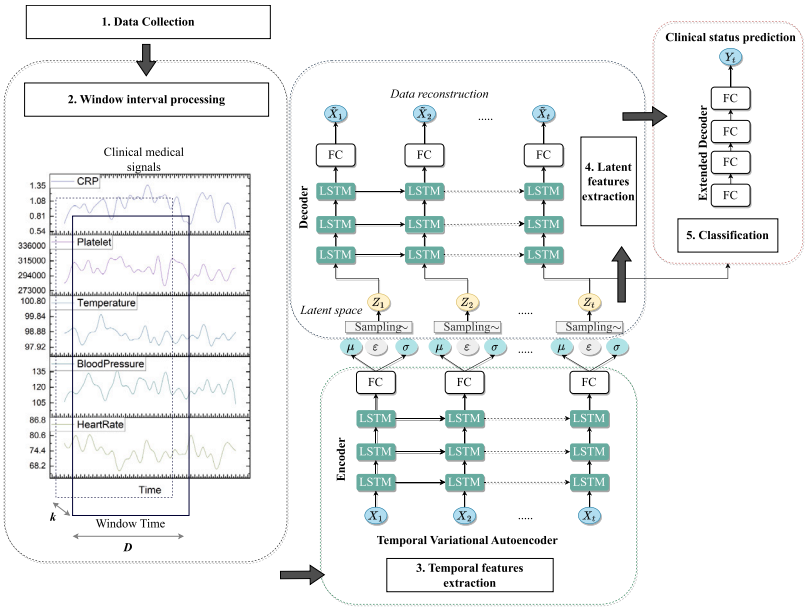


Fig. 2. The overall framework of the proposed TVAE system. The main function of this system is to determine whether a patient's condition at a time point is event or non-event (Y) based on the probability of an anomaly occurring (DEWS score), from input data (X). D indicates window time size; k: window sliding step size; LSTM: Long-Short Term Memory; FC: Fully Connected Layer; μ, σ and ε are mean, variance and gaussian distribution, respectively. TVAE is the combination of a Variational Auto-Encoder (VAE) [33] and LSTM [34]. Similar to other VAE network architectures, our model consisted of an encoder and decoder path, as well as an extended encoder for the temporal classification task. We used the LSTM encoder for the time-series data and took advantage of VAE for translating the input into the latent space features and for converting them into textural data. The TVAE model has two decoders that serve different purposes. The LSTM decoder performed the reconstruction task. This stage decoded the latent features and transformed the LSTM encoder into the original type of input data so that the latent representations were accurate and consequential. The FCN decoder was applied for the clinical prediction task - the main task of DEWS, using the latent representation as input.

3.2. Method

This study develops a DEWS that can predict a patient's abnormality at a certain time based on vital signs and laboratory test records measured in the past (time-series data). This whole problem is considered a classification task where a patient's status at an input time is non-event or event (abnormal status). Data imbalance is a dilemma encountered by the system, in which the minority class (event class) accounts for only an extremely small percentage of the entire data (less than 1%). Our goal is to ensure "urgent" - return results in the nearest time and high "sensitivity" - to minimize the false predictions for the minority class (events).

Our proposed system is illustrated in Fig. 2, comprising five main steps: Data Collection, Window Interval Processing (WIP), Temporal Features Extraction, Latent Features Extraction, and Classification. After performing data collection, we focus on the problem of missing values in the natural medical data filling them with a median filter. Missing patient vital signs and laboratory tests are determined by the mean of the non-missing values for each patient's index class. Then, the input temporal data, comprising measurement indexes of the patient on each timestamp, are normalized and processed in sequenced window time based on the WIP stage. We framed the clinical status prediction task as a binary classification problem, where the input was the window time $D = [x_1; ...; x_t]$ of the size t, containing the input clinical variables (vital signs, laboratory tests) of a patient on t time steps. The output of the TVAE was the DEWS score (in the range of 0 to 100), which illustrated the abnormal/event probability of the patient. Through an LSTM encoder, TVAE took input window time and produced the temporal presentation. Then, temporal features are put into latent space to produce latent representation based on their mean and covariance. An LSTM decoder with a similar architecture is used to construct the input variables and validate the effectiveness of latent representation. Thereby increasing the robustness of the latent representation. Finally, A drop-out-based FC layer, proposed as an extended decoder, used the latent feature as input for producing a deep early warning system score (DEWS score). A co-training pipeline is applied between the two decoders to improve the performance of the latent features and the clinical status prediction task.

3.2.1. Data collection

Considering a time step as a sample, we collected a total of 317,006 samples from 2615 patients' records in CNUH and a total of $2,217,958$ samples from 8105 patients in UV. We used all CNUH samples for the experiment. However, the total samples of 6738 non-event patients, which was the UV number, were excluded because of the limitation of our resources for training. Nevertheless, the number of non-event samples is still larger than the event group. Specifically, we have a total of 317,006 time points and 1042 event time points for CNU and 493,333 time points with only 23,881 event time points for UV.

3.2.2. WIP

Data pre-processing is a DL technique used to transform raw data into meaningful and understandable formats [35,36]. We scaled the input data x with standard normalization because of the multivariable properties and the various features values scaling on our framework $\overline{x} = \frac{x-\mu}{\sigma}$, where μ and σ were the mean and standard deviation of the input data, respectively.

We processed a sliding window mechanism to initialize the window time and smoothly fed it into the TVAE. We truncated the measurement variables of the first D time step for each patient to create sequence input for the temporal model and presented the window time. TVAE predicts the patient's event probability y_t at the final time point of D based on the information of the previous D timestamp on the window time. Next, with the sliding step size k, the sliding window mechanism makes the window time slide through all the time points of each object and predicts the output of each patient for every next k timestamp. Thus, we performed different experiments to select the optimal indicators by varying the parameters of the window size D and sliding step size k. However, this depends on the system requirements and the condition of the data. This shows the system flexibility since we can customize the input and output data size based on two parameters: D and k.

3.2.3. Temporal features extraction

After the WIP stage, our dataset is formed as $\mathbb{D} = \{D_1, \ldots, D_N\}$, where N corresponding total number of samples (window time) that are generated from the truncate step. Each window time $D = [[x_{ij}, t_i]_{j=1}^{n}]_{i=1}^{T}$ contains total n clinical variables of t time steps. Hence, we determine the input temporal data as a sequence with the shape of $[N, t, n]$.

First, we use an RNN-LSTM structured encoder to extract temporal features from input sequences. RNN is the most common structure for time series data. The RNN loop structure helps to process the sequential data formed as $[N, t, n]$. This repetition allows previous information to affect the current task. This process is similar to a doctor's checking the patient's clinical history for analyzing the patient's condition. However, the influence of initial information could be reduced using vanilla RNN as the dimension of the series increases. This was known as the "long-term dependency" problem [37]. Thus, LSTM avoids this issue through gates to hold information prematurely. Therefore, we model the LSTM structure in the encoder ($q_\phi(z|x)$) and decoder $p_\theta(x|z)$ of VAE. In this model, the encoder consists of 3 LSTM layers, in which the number of hidden units are 100/50/25, respectively. For each element in each input window time, we updated the hidden state $h_t, t \in [1, T]$ as follows:

$$\begin{aligned} f_t &= \sigma\left(W_{f,x}x_t + W_{f,h}h_{t-1} + b_f\right) \\ \tilde{s}_t &= \tanh\left(W_{\tilde{s},x}x_t + W_{\tilde{s},h}h_{t-1} + b_{\tilde{s}}\right) \\ i_t &= \tanh\left(W_{i,x}x_t + W_{i,h}h_{t-1} + b_i\right) \\ s_t &= f_t \circ s_{t-1} + i_t \circ \tilde{s}_t \\ o_t &= \sigma\left(W_{o,x}x_t + W_{o,h}h_{t-1} + b_o\right) \\ h_t &= o_t \circ \tanh\left(s_t\right) \end{aligned} \tag{1}$$

where x_t is the first vector at each time step t; $W_{f,x}$, $W_{f,h}$, $W_{\tilde{s},x}$, $W_{\tilde{s},h}$, $W_{i,x}$, $W_{i,h}$, $W_{o,x}$ and $W_{o,h}$ are the weight matrices in each LSTM cell; b_f, $b_{\tilde{s}}$, b_i and b_o are bias vectors; f_t, i_t and o_t contains the activation values for the forget gate, input gate and output gate, respectively; s_t and $\tilde{s}_t$ are representation vector for cell internal state and candidate value, respectively; and h_t - the temporal features of LSTM cell, is the hidden state at time t.

3.2.4. Latent features extraction

After generated, h_t is feed into 2 linear layers to compute mean (μ_t) and log-variance ($log\sigma_t$) of the posterior $p(z_t|x_t)$. Let us denote the weights and biases of these layers as W_μ, b_μ for the mean and W_σ, b_σ for the log-variance. The computational function can be expressed as follows:

$$\mu_t = W_\mu h_t + b_\mu \tag{2}$$

$$\log \sigma_t = W_\sigma h_t + b_\sigma \tag{3}$$

Here, W_μ, b_μ, W_σ, and b_σ are the learnable parameters of the linear layers. The output of these layers provides the parameters for the Gaussian distribution representing the posterior $p(z_t|x_t)$. We added a Gaussian distribution ϵ for sequential data. We also sampled μ_t, $log\sigma_t$ and ε to produce latent representation z_t by a sampling function:

$$z_t = \mu_t + e^{0.5 \cdot log\sigma_t} \cdot \varepsilon \tag{4}$$

Here, we take advantage of the VAE structure to increase the robustness of the latent presentation through reconstruction training.

The VAE was introduced as an alternative form of auto-encoder (AE) derived from Bayesian algorithm [33]. In VAE, each input data x was represented by an unobserved latent feature z which was extracted by the normal distribution. For our model, the encoder-to-decoder structure applies to reconstruct the input data to validate z's ability to learn the probability distribution between classes. In which, the encoder $q_\phi(z|x)$ estimates the actual consequences and the decoder $p_\theta(x|z)$ represents the likelihood of complex data reconstruction progress that returns x from z. In other words, the more similar the output from $p_\theta(x|z)$ is to the input value, the better the latent feature z is. For this, we used the Kullback–Leibler (KL) divergence function to quantify the distance between $q_\phi(z|x)$ and $p_\theta(z)$:

$$D_{KL}\left[q_\phi(z \mid x) \parallel p_\theta(z)\right] = E_{z\sim p_\theta}(z \mid x)\left[\log p_\theta(z \mid x) - \log q_\phi(z \mid x)\right] \tag{5}$$

In which, this function could be modified after applied Bayesian theorem:

$$\log p_\theta(x) = D_{KL}\left[q_\phi(z \mid x) \parallel p_\theta(z)\right] + L(\theta, \phi; x), \tag{6}$$

$$L(\theta, \phi; x) = E_{z\sim q_\phi(z|x)}\left[-\log q_\phi(z \mid x) + \log p_\theta(x \mid z) + \log p_\theta(z)\right] \tag{7}$$

$L(\theta, \phi; x)$ was known as the evidence lower bound (ELBO). The VAE could optimize the parameters θ, ϕ by maximizing lower bound of the probability of generating original data samples.

$$L_{vae} = L(\theta, \phi; x) \leq \log p_\theta(x) \tag{8}$$

We improved the quality of latent feature z by reconstructing data to minimize the loss function Eq. (8).

3.2.5. Classification through extended decoder based drop out FCN

After optimizing the distributional representation of latent features z_t, we fed it into an extended encoder dedicated to time classifiers to predict the patient's clinical deterioration. We proposed a decoder using drop-out based on a Fully Connected Network (FCN) of 4 FC layers. This decoder consists of 4 FC layers with 8/64/32/16 hidden units, respectively to produce event/abnormal probability $p(y_t)$, corresponding to x_t on the total N samples. For this task, the clinical prediction loss was defined based on Binary-cross entropy loss:

$$L_{clinical} = -\frac{1}{N}\sum_{i=1}^{N} y_t \cdot \log\left(p\left(y_t\right)\right) + \left(1 - y_t\right) \cdot \log\left(1 - p\left(y_t\right)\right) \tag{9}$$

Our key task is to overcome the imbalance problem. A major bias problem is to train a DL model in a dataset characterized using a significant numerical difference between the majority class and minority class, especially when the model achieves high accuracy for predicting the majority class. Our experiment using the CNUH data shows that only 1.5% of the total population are event/abnormal patients. This shows that our model consistently achieves high accuracy and offers good performance in predicting the normal patient class. However, when evaluating the prediction accuracy of the model on the event class, the system performed a disappointing result with a high false prediction rate. In some cases, the late alarm rate reaches 40% (80 of 200 event samples are predicted as normal samples). With this number of late alarm rates, it is challenging to integrate DEWS in real-time TTSs and RRS. For this, we applied an imbalance score based on Dice loss [38,39] implemented on our data as follows:

$$S_i = \frac{2 \cdot \text{True Positives}}{2 \cdot \text{True Positives} + \text{False Positives} + \text{False Negatives}} \tag{10}$$

where S_i is the imbalance score, True Positives represent the number of instances where the model correctly predicts the positive class (event). False Positives represent the number of instances where the model incorrectly predicts the positive class (event) when the true class is negative (non-event). False Negatives represent the number of instances where the model incorrectly predicts the negative class (non-event) when the true class is positive (event). For a typical sample x_t, function Eq. (10) is formulated as follows:

$$S_i(x_t) = \frac{2p_t y_t}{p_t + y_t} \tag{11}$$

Besides, we add η as the smoothing factor ($\eta = 1$ in our experiment) to both numerators and denominators. In this case, $\frac{\eta}{p_t + y_t + \eta}$ is seen as the negative (normal) samples. The peculiarity of this function was the ability to emphasize the event or abnormal patient class and prevent bias problems. For easier optimization, the final imbalance loss was as follows:

$$L_i = 1 - \frac{2\sum_t p_t y_t + \eta}{\sum_t p_t^2 + \sum_t y_t^2 + \eta} \tag{12}$$

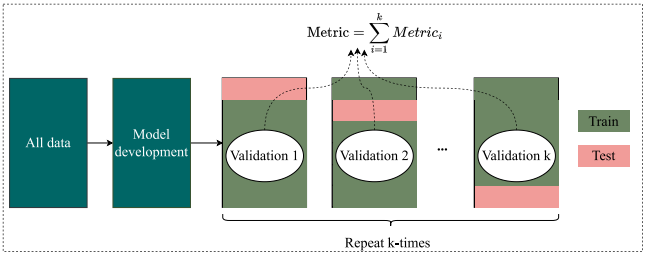
In the training process, the reconstruction and clinical deterioration tasks were performed individually. However, these two processes complement to improve the performance of the system. For this, the reconstruction loss (L_{vae}) was updated through classification losses. By backpropagating the losses of the classification task to VAE, the VAE improved parameters optimization for producing a worthy latent representation and for predicting the clinical deterioration task. Therefore, the final loss is defined as:

$$L_{final} = L_{vae} + L_{clinical} + L_i \tag{13}$$

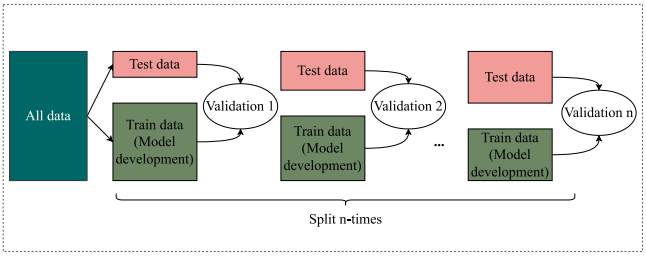
4. Results and discussion

4.1. Experiment setting

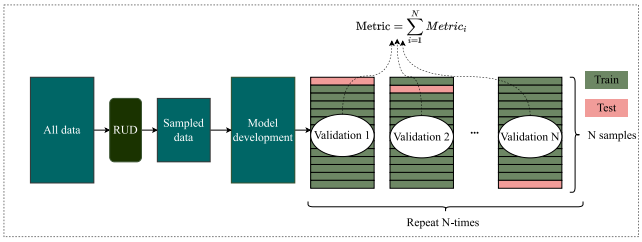
All computations of this study were performed using Python and TensorFlow (v. 2.11.0) on an Nvidia GeForce RTX 3090 graphics processing unit (driver v. 530.30.02) and CUDA (v. 12.1) with 64 GB memory. As discussed in previous sections, we trained and evaluated the CNUH and UV datasets independently because of the differences in the size of their features and labels. As WIP is an integral component of our system process, the sample count reports in this section pertain to pre-WIP statistics. The results sample size in the examined cases may fluctuate based on the variant values of the variables D and k.



(a) K-Fold CV.



(b) VTSA.



(c) LOOCV.

Fig. 3. Validation methods description. Fig. 3(a): K-Fold CV. Fig. 3(b): VTSA. Fig. 3(c): LOOCV. $Metric$ - overall evaluation metrics of the models, $Metric_i$ - evaluation metrics of a single CV fold. RUD: Random Under-Sampling. N: total number of samples in the dataset. In K-Fold CV, the dataset is divided into k equally-sized folds. The model is trained and validated k times, each time using a different fold as the test set and the remaining data as the training set. The final evaluation metric is averaged (95% CI) over all iterations. In VTSA, we perform n experiments based on n times stratified train/test split data with variation of split size values. LOOCV partitions the data into folds for cross-evaluation. Nevertheless, in each iteration, a single data point serves as the validation set, while the remaining data are utilized for training.

4.1.1. Validation strategies

In order to rigorously assess the robustness and generalization capability of our model, we employed three main validation approaches: K-Fold Cross-Validation (K-Fold CV), Variation Test Sensitivity Analysis (VTSA), and Leave-One-Out Cross-Validation (LOOCV). The detail information of main validation methods are described in Fig. 3.

K-Fold CV (Fig. 3(a)), a popular technique in machine learning [40], was used to partition our datasets into multiple training and validation sets, ensuring a comprehensive assessment. In K-Fold CV, we evaluate the proposed method based on three criteria: evaluation of model performance across multiple study cases, ablation study for imbalance loss, and validation of the efficiency of hidden representation through T-SNE visualization [41].

We conducted VTSA (Fig. 3(b)), a method designed to examine the model's performance under variations in data size, providing insights into its efficiency with varying training and testing data proportions. VTSA aims to validate model performance independently of the influences stemming from variations in data quality and quantity which exert distinct impacts on model effectiveness [42,43].

During the monitoring process of RRS, the size and quality of the data can be influenced by various unforeseen external factors [28]. This introduces potential biases and limitations associated with small sample sizes, such as uncertain and biased performance estimates [29,30].

Table 2
Summary of the architecture and hyperparameter settings of the comparison models. Conv1D - 1D Convolution Layer, ReLU - Rectified Linear Unit activation function layer, BN - BatchNormalization, GAP - Global average pooling 1D layer. Some related studies in comparative methods do not provide detailed information about the model or source code. Therefore, we reimplement these models based on the descriptions in the corresponding related study and select detailed parameters based on our hyperparameter tuning method. The values in the table briefly describe the structure and parameters of the DL models. For instance, DCNN have 5 hidden layers including 1 FC layer with 128 hidden units; 2 Conv1D layers with 32 and 64 hidden units, respectively, and 2 Max Pooling 1D layers with 32/64 hidden units, respectively. All activation function in the hidden layers of DCNN is ReLU, Sigmoid is the activate function in the classification step; dropout rates and learning rate is 0.5 (for FC layer) and 0.0001, respectively. DCNN using Adam optimization. Total training params = 95,410.

Model		TVAE	BiLSTM + Attention [21]	RNN [20]	FCN	DCNN [17]	FCNN [18]
Layers (number of hidden units)	Number of hidden layers	7	7	3	4	5	10
	LSTM	100/50/25	–	100/50/25	–	–	–
	FC	8/64/32/16	100/50/25	–	8/64/32/16	128	–
	BiLSTM	–	100/50/25	–	–	–	–
	Conv1D	–	–	–	–	32/64	64/64/64
	ReLU	–	–	–	–	–	64/64/64
	BN	–	–	–	–	–	64/64/64
	MaxPool1D	–	–	–	–	32/64	–
	GAP	–	–	–	–	–	64
	Attention	–	10	–	–	–	–
Hyperparameter setting	Hidden layers activation	ReLU	ReLU	Tanh	ReLU	ReLU	–
	Classification head activation	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Sigmoid
	Dropout rates	0.2/0.2/0.2/0.1	0.2	0.2/0.2/0.1	0.2/0.2/0.2/0.1	0.5	–
	Learning rate	0.0001	0.0001	0.0001	0.0001	0.0001	0.0001
	Optimizer	Adam	Adam	Adam	Adam	Adam	Adam
	Total params	91,906	102,338	88,252	58,274	95,410	103,466

Therefore, we demonstrate chance-level variation results together with LOOCV. LOOCV (Fig. 3(c)) is an alternative form of K-Fold CV where each fold will be a sample out of a total of N samples of the data. In this study, the magnitude of the original data size does not allow LOOCV to perform on the entire data. Therefore, before the prediction step, we perform Random Under-Sampling (RUD) [44] to balance the number of labels between the two classes and reduce the data size. After RUD, the class imbalance is resolved but the problem is faced with a small amount of data.

4.1.2. Comparison methods and model's parameters configuration

Based on related approaches discussed in Section 2, we use three types of structures for comparison.

- ML model: XGBM [16] - method shows high efficiency among ML models.
- RNN structure models:
  - RNN: implemented based on the description of the study [20] - 3 LSTM layers with 100/50/25 hidden units, respectively.
  - BiLSTM + Attention: implemented based on the description of the study [21] - 1 Bidirectional (100 hidden units) + Attention block.
- Non-RNN structure models:
  - DCNN: implemented based on the description of the study [17] - 2 Conv1D layers with 32/64 filters, respectively, followed by MaxPooling1D layers for downsampling. The subsequent Flatten layer extracts features, and the FC with 128. A dropout layer with a dropout rate of 0.5 is applied before the classification.
  - FCNN: implemented based on the description of the study [18] - 3 Conv1D layers with 64 filters each, employing batch normalization and ReLU activation. The global average pooling layer is applied to capture spatial information before the classification.
  - FCN: Has the same architect of the decoder of our TVAE model with 4 Dense drop-out layers with 192/96/48/24 hidden units, respectively.

Table 2 shows the architecture and hyperparameter settings of comparison DL models. The parameters shown in this table are the optimal parameters selected from the hyperparameter tuning and searching process. To determine optimal hyperparameter values for the DL models, we employed KerasTuner [45], an effective hyperparameter optimization library integrated with the Keras DL framework. This approach utilizes a search space defined by the user, exploring various combinations of hyperparameter values to identify the configuration that achieves the best model performance.

The system's hyperparameter tuning progress to get the best configuration H_M for a DL model M includes these steps:

- Search space definition (S_M): Select a range of values and distributions for each hyperparameter relevant to M. S_M = {dropout rates, layer units, learning rate, batch size, ...}
- Objective function (O_M): Specify an objective function specific to M that quantifies its performance based on a chosen metric (e.g., AUROC, F1).
- Perform search algorithms (A_M): In this step, we implemented two approaches: Random Search [46] and Bayesian Optimization [47].
- Evaluation (E_M): Automate the process of training and evaluating models for different hyperparameter combinations according to the chosen objective function.
- Best parameters configuration (H_M): From the search progress, we got the optimal parameters for the DL models. The overall hyperparameters tuning process can be expressed as:

$$H_M = A_M\left(E_M\left(O_M\left(S_M\right)\right)\right) \tag{14}$$

In simplest terms, H_M is derived by applying the chosen search algorithm A_M to the automated evaluation process E_M, which is led by the objective function O_M and the stated search space S_M.

For XGBM, we use the popular hyperparameter method for ML models, GridsearchCV of the scikit-learn library, to achieve optimal parameters.Unlike TVAE, we applied only the clinical loss and imbalance loss to these models. We trained all models with 1000 epochs, batch size = 2048. The stopping condition was when L_{final} converges.

4.1.3. Evaluation metrics

In this study, for imbalance evaluation, we use AUROC, AUPRC, and Late Alarm (also known as type II error) to provide insights into its

ability to identify positive instances. Besides, Accuracy (%) is used to present the chance-level results in LOOCV, calculated as:

$$\mathrm{AC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FN} + \mathrm{TN} + \mathrm{FP}} \tag{15}$$

where TP, FP, TN, and FN represent True/False predictions for Event /Non-event labels, which are used to explain the classification technique's various results [48]. The Cohen's Kappa score is employed to quantify the agreement between observed accuracy (AC) and chance (P_e) [49], and it is calculated as:

$$\begin{cases} P_e = \frac{(\mathrm{TP}+\mathrm{FP})(\mathrm{TP}+\mathrm{FN})+(\mathrm{TN}+\mathrm{FP})(\mathrm{TN}+\mathrm{FN})}{(\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN})^2} \\ \mathrm{Kappa} = \frac{\mathrm{AC}-P_e}{1-P_e} \end{cases} \tag{16}$$

We also employ the F1 Score as a crucial evaluation metric to comprehensively assess the model's ability to accurately classify positive cases. The F1 Score is calculated using the formula:

$$F1 = 2 \times \frac{\mathrm{Precision} \times \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}} \tag{17}$$

4.2. K-fold CV

We applied 5-fold cross-validation with a constant seed (seed = 42) and reported the mean comparison metrics of 5 folds. Thus, every set shows a high imbalance rate, with around 0.3% on CNUH and 5.5% on the UV dataset. Due to that imbalance, we used the AUROC to distinguish the positive from negative samples for the binary classification task. Additionally, we validated our model's performance with AUPRC, used as an alternative to AUC for solving problems with a large skew in the class distribution [50]. We computed the late alarm, defined as the number of samples that the system predicted as a non-event, to assess the model's applicability. However, they are event-time steps. Thus, the lower the late alarm, the better the model. This index assesses the sensitivity and "urgency" of a model in detecting the abnormal status of the patient. Besides, probability sample results are also presented to evaluate the comparison methods visually and realistically. Besides, F1 Score and Kappa Score are useful metrics for evaluating model performance on imbalanced datasets because they consider both false positives and false negatives, providing a balanced assessment of the classifier's effectiveness. We then use the T-SNE visualization method to evaluate the separation between the two layers (non-event and event) of latent features compared with other methods. Besides, a sub-experiment was also implemented to evaluate the effect of imbalance loss (L_i) on the output (the ablation study).

During processing, the size of the output of this stage changes, and this depends on how we customize the window size (D) and window step size (k) parameters. We experimented with a variety of k and D cases. However, based on the characteristics of the available data for physician consultations and the availability of resources devoted to the study, we performed experiments in the following cases: (1) sliding step size = 1 h ; window size = 8/16/24 h - this showed predicting the clinical status for the next 1 h using the corresponding clinical information for the last 8/16/24 h. (2) Window size = 8 h; sliding size = 8/10/16 h - this shows that based on clinical information in the previous 8 h, we can predict clinical status in the next 8/10/16 h. For case (1), the information stored for prediction must not exceed the total number of times for a patient, in which the least number of times steps a patient has been in the hospital is 24 h (hospital stay only for 1 day). Therefore, the maximum window time size is 24 h. According to the doctors' advice, 8 h is the minimum window size required for prediction. For case (2), the future time points for prediction are chosen to match the response team's ability.

4.2.1. Study case 1 - predict next 1 h; customize previous information by 8/16/24 h

The AUROC, AUPRC, and Late Alarm experimental results on CNUH and UV are shown in Tables 3 and 4. The average Kappa and F1 results over 5 folds of the two datasets are shown on Table 5. For 2615 patients on CNUH, our proposed TVAE outperforms all RNN-based approaches in 3 options of window size. As the window size increases, the models obtain better results and increase the effectiveness of clinical predictions. With D = 24 h, TVAE obtains the best performances with 0.973 of AUROC, 0.887 of AUPRC, 0.781 of Kappa, 0.942 of F1 score, and reduces up to 76.2 late alarm cases compared with other methods. XGBM and FCNN have higher AUPRC results in this option D = 16 h. However, these models give more late alarms (higher up to 7.8 of late alarm sample compared to TVAE). This shows that they are still suffering from bias due to a high imbalance problem.

For UV, TVAE achieves the best results except for option D = 16 h. In this case, our method has the highest AUROC index (AUROC = 0.989) but has 0.076 of AUPRC reduction and gets a later alarm at 19.4 samples compared with the RNN. Among the comparison methods, the BiLSTM + Attention model performed much worse than the other methods. This suggests that the structure of the Attention mechanism is unsuitable for our WIP or could not interpret our datasets.

Fig. 4 shows the probability results of some samples of some comparison models. In all samples in this Figure, our method still shows more stability than the compared methods, with the most returning high event probability between *detection time* and *event time* (beyond the specified threshold for label selection). BiLSTM + Attention (Fig. 4(a)), FCN, and RNN (Fig. 4(b)) seem to be equivalent to TVAE in the 'event interval'. However, for non-event time points, these models still return a very high event probability, leading to false alarms. For TVAE, this is very rare.

4.2.2. Study case 2 - predict the next 8/10/16 h; using the previous 8 h information

The AUROC, AUPRC, and Late Alarm experimental results on CNUH and UV are shown in Tables 6 and 7. The average Kappa and F1 results over 5 folds of the two datasets are shown on Table 8. In the patient's event prediction of the next 8/10/16 h task, our approach shows more improvements in results than other comparative methods on 2 datasets, despite a slight decline in performance compared with the study case 1. Increasing the sliding step size while keeping the same window time size is a challenging task for clinical status since we are unsure that the information from the previous hours is sufficient to transmit and determine the result of a time point in the distant future. However, in CNUH, our method still performs most excellently in predicting clinical status at the furthest time (16 h). This shows that the ability to store and transmit information from the LSTM structure has brought certain benefits to our system. For UV, our model has the best performance with k = 10. However, this result seems degrading by increasing the sliding step size by 16 h. The reason for this difference is the data structure of each data set. The sliding step size is too large for the total number of time steps of the 'event interval' and will cause the window time to skip some important time points in this interval. This is an existing problem that we must improve. The probability results of 2 samples of some models in this case are illustrated and explained in Fig. 5.

4.2.3. Performance of feature space representation

Fig. 6 could validate the outperformance of TVAE compared to other methods. The reason for these impressive results was the appreciable contribution of TVAE's compact latent features. The outstanding performance of our TVAE model shows the distinguished ability of latent representation for learning the class distribution effectively, thanks to local and global dependent properties discoverability of LSTM and VAE. We used t-SNE to illustrate the performance of feature space representation in the comparison methods on the UV dataset. The

Table 3
Experiment results on CNUH dataset. Custom window size (D) = [8; 16; 24]; sliding step size k = 1. Predict the next 1 hour's clinical status.

Window time size (D)	Model	Metric (95% CI)		
		AUROC	AUPRC	Late alarm
8	BiLSTM + Attention [21]	0.799 (0.770 – 0.827)	0.572 (0.517 – 0.627)	80.8 (66.8 – 94.7)
	FCN	0.898 (0.885 – 0.911)	0.693 (0.658 – 0.728)	40.6 (35.5 – 45.6)
	RNN [20]	0.871 (0.854 – 0.888)	0.714 (0.687 – 0.741)	51.6 (42.3 – 60.8)
	DCNN [17]	0.902 (0.879 – 0.926)	0.734 (0.691 – 0.886)	37.6 (28.9 – 46.2)
	FCNN [18]	0.921 (0.905 – 0.937)	0.800 (0.777 – 0.823)	30.4 (24.0 – 36.7)
	XGBM [16]	0.955 (0.946 – 0.963)	0.811 (0.801 – 0.829)	18.0 (14.5 – 21.4)
	TVAE	**0.970 (0.958 – 0.982)**	**0.829 (0.752 – 0.906)**	**11.6 (6.8 – 16.3)**
16	BiLSTM + Attention [21]	0.843 (0.801 – 0.886)	0.647 (0.529 – 0.765)	59.2 (45.1 – 73.2)
	FCN	0.917 (0.895 – 0.939)	0.763 (0.712 – 0.813)	31.4 (22.6 – 40.2)
	RNN [20]	0.950 (0.928 – 0.981)	0.846 (0.794 – 0.899)	16.6 (7.7 – 25.5)
	DCNN [17]	0.944 (0.917 – 0.972)	0.811 (0.763 – 0.859)	20.6 (12.1 – 29.0)
	FCNN [18]	0.953 (0.934 – 0.973)	0.863 (0.827 –0.899)	17.8 (9.6 – 26.0)
	XGBM [16]	0.960 (0.945 – 0.974)	0.911 (0.870 – 0.953)	15.2 (10.1 – 20.2)
	TVAE	**0.970 (0.940 – 0.990)**	**0.858 (0.809 – 0.906)**	**10.8 (2.4 – 19.2)**
24	BiLSTM + Attention [21]	0.759 (0.575 – 0.944)	0.496 (0.167 – 0.826)	85.8 (26.1 – 145.4)
	FCN	0.910 (0.894 – 0.825)	0.758 (0.736 – 0.781)	32.6 (25.7 – 39.4)
	RNN [20]	0.962 (0.936 – 0.987)	0.857 (0.801 – 0.913)	13.8 (4.0 – 23.6)
	DCNN [17]	0.949 (0.929 – 0.969)	0.844 (0.797 – 0.892)	18.2 (11.1 – 25.3)
	FCNN [18]	0.957 (0.933 – 0.981)	0.866 (0.825 – 0.908)	15.4 (6.6 – 24.1)
	XGBM [16]	0.960 (0.941 – 0.979)	0.814 (0.805 – 0.923)	14.6 (7.2 – 21.9)
	TVAE	**0.973 (0.955 – 0.989)**	**0.887 (0.846 – 0.927)**	**9.6 (3.5 – 15.6)**

Table 4
Experiment results on UV dataset. Custom window size (D) = [8; 16; 24]; sliding step size k = 1. Predict the next 1 hour's clinical status.

Window time size (D)	Model	Metric (95% CI)		
		AUROC	AUPRC	Late alarm
8	BiLSTM + Attention [21]	0.823 (0.593 – 0.998)	0.744 (0.552 – 0.935)	1604.6 (546.3 – 3755.5)
	FCN	0.898 (0.885 – 0.911)	0.830 (0.808 – 0.851)	225.0 (180.7 – 269.3)
	RNN [20]	0.969 (0.962 – 0.975)	0.890 (0.881 – 0.899)	263.6 (212.3 – 314.8)
	DCNN [17]	0.930 (0.920 – 0.940)	0.808 (0.793 – 0.824)	621.0 (523.9 – 718.0)
	FCNN [18]	0.974 (0.970 – 0.979)	0.919 (0.912 – 0.927)	221.0 (172.9 – 269.0)
	XGBM [16]	0.907 (0.906 – 0.908)	0.803 (0.800 – 0.807)	842.4 (831.5 – 853.2)
	TVAE	**0.983 (0.976 – 0.990)**	**0.909 (0.906 – 0.913)**	**136.4 (68.9 – 203.8)**
16	BiLSTM + Attention [21]	0.947 (0.935 – 0.960)	0.863 (0.854 – 0.872)	437.0 (327.1 – 546.8)
	FCN	0.986 (0.982 – 0.991)	0.923 (0.919 – 0.926)	103.0 (70.5 – 135.4)
	RNN [20]	0.969 (0.962 – 0.975)	0.991 (0.990 – 0.992)	63.0 (58.4 – 67.5)
	DCNN [17]	0.959 (0.944 – 0.974)	0.880 (0.859 – 0.900)	343.4 (214.3- 472.4)
	FCNN [18]	0.980 (0.974 – 0.986)	0.912 (0.903 – 0.954)	165.8 (110.8 – 220.7)
	XGBM [16]	0.922 (0.916 – 0.927)	0.835 (0.822 – 0.848)	680.6 (641.8 – 719.3)
	TVAE	**0.989 (0.987 – 0.991)**	**0.915 (0.904 – 0.925)**	**82.4 (55.4 – 109.3)**
24	BiLSTM + Attention [21]	0.946 (0.922 – 0.971)	0.867 (0.842 – 0.891)	431.6 (225.2 – 637.9)
	FCN	0.991 (0.988 – 0.993)	0.927 (0.922 – 0.931)	65.8 (45.0 – 86.5)
	RNN [20]	0.991 (0.99 – 0.993)	0.920 (0.914 – 0.925)	55.2 (44.0 – 66.3)
	DCNN [17]	0.980 (0.977 – 0.983)	0.915 (0.913 – 0.917)	155.6 (129.4 – 181.7)
	FCNN [18]	0.986 (0.979 – 0.992)	0.922 (0.92 – 0.924)	109.4 (50.9 – 167.8)
	XGBM [16]	0.938 (0.934 – 0.941)	0.868 (0.861 – 0.875)	523.4 (497.1 – 549.7)
	TVAE	**0.992 (0.988 – 0.996)**	**0.926 (0.922 – 0.93)**	**51.6 (13.1 – 90.0)**

brown point presents the number of abnormal samples. Blue denotes normal samples. For our TVAE model, we presented the latent space feature z; for other methods, we visualized the final representation before the classification (sigmoid) layer. However, our approach has a border large enough to separate 2 classes among the comparison methods. This led to a remarkable reduction in the late alarm rate. Additionally, the DEWS scores sample results show the sensitivity of our model's DEWS score, thanks to the model's ability to separate the event and non-event classes. The combination of LSTM, VAE, and LSTM can discover local dependent properties. Meanwhile, VAE is for discovering global properties in the data distribution, which assists our framework in increasing interpretative learning for temporal data.

4.2.4. Ablation study for imbalance loss

In this experiment, we predict patients' clinical deterioration in future time periods (next 8 h, 10 h, and 16 h) using TVAE with and without Imbalance Loss. Naturally, It is reasonable to expect that the performance of the clinical impairment prediction model will decrease as the prediction interval is lengthened. The reason is that the further into the future the predicted period, the more uncertain the patient's clinical condition and the more factors that can influence their clinical course. However, the results in Figs. 7(a) and 7(b) give results different from the above expectation. When experimenting with the UV dataset (Figs. 7(c) and 7(d)), the results when applying Imbalance Loss are also completely superior. The AUROC and Late Alarm results when using Imbalance Loss increase despite the expansion of the future prediction period showing our proposed method has been effective for predicting clinical deterioration. By incorporating Imbalance Loss in the model training process, the system could be better able to identify patterns and features in the input biomedical data that are associated with the risk of clinical deterioration, even if such patterns are relatively rare or occur only in a small subset of patients. This could help the model maintain predictive performance over a longer period of time, by allowing the model to identify subtle changes in a patient's clinical condition that could be indicative of risk. Besides, the application of multi-task learning with Imbalance Loss, Clinical Loss, and VAE Loss helps our model avoid bias and insensitivity of the DEWS score - a

Table 5
Kappa and F1 scores on two dataset. Custom window size (D) = [8; 16; 24]; sliding step size k = 1. Predict the next 1 hour's clinical status.

Dataset		CNUH		UV	
Window time size (D)	Model	Kappa	F1	Kappa	F1
8	BiLSTM + Attention [21]	0.610	0.735	0.566	0.683
	FCN	0.689	0.831	0.790	0.952
	RNN [20]	0.689	0.831	0.782	0.943
	DCNN [17]	0.708	0.854	0.742	0.894
	FCNN [18]	0.740	0.892	0.794	0.957
	XGBM [16]	0.750	0.904	0.722	0.870
	TVAE	**0.754**	**0.909**	**0.798**	**0.962**
16	BiLSTM + Attention [21]	0.658	0.793	0.757	0.913
	FCN	0.723	0.872	0.806	0.972
	RNN [20]	0.762	0.919	0.807	0.973
	DCNN [17]	0.747	0.900	0.776	0.935
	FCNN [18]	0.771	0.929	0.804	0.969
	XGBM [16]	0.766	0.923	0.752	0.907
	TVAE	**0.768**	**0.926**	**0.811**	**0.977**
24	BiLSTM + Attention [21]	0.508	0.613	0.759	0.915
	FCN	0.721	0.869	0.810	0.976
	RNN [20]	0.767	0.925	0.806	0.971
	DCNN [17]	0.762	0.918	0.801	0.965
	FCNN [18]	0.772	0.930	0.707	0.973
	XGBM [16]	0.792	0.955	0.766	0.923
	TVAE	**0.781**	**0.942**	**0.811**	**0.978**

Table 6
Experiment results on CNUH dataset. Window size (D) = 8. Custom window step size k = [8; 10; 16]. Predict the next 8/10/16 hour's clinical status.

Window step size (k)	Model	Metric (95% CI)		
		AUROC	AUPRC	Late alarm
8	BiLSTM + Attention [21]	0.793 (0.742 – 0.845)	0.543 (0.387 – 0.698)	73.6 (57.8 – 89.3)
	FCN	0.895 (0.875 – 0.915)	0.701 (0.636 – 0.764)	38.8 (34.2 – 43.3)
	RNN [20]	0.875 (0.852 – 0.899)	0.718 (0.671 – 0.765)	44.4 (37.1 – 51.6)
	DCNN [17]	0.913 (0.878 – 0.948)	0.717 (0.693 – 0.761)	31.0 (19.0 – 43.0)
	FCNN [18]	0.920 (0.881 – 0.958)	0.713 (0.689 – 0.837)	28.6 (16.8 – 40.4)
	XGBM [16]	0.921 (0.901 – 0.931)	0.716 (0.670 – 0.802)	27.6 (13.4 – 21.8)
	TVAE	**0.928 (0.910 – 0.947)**	**0.720 (0.569 – 0.872)**	**25.2 (19.0 – 31.3)**
10	BiLSTM + Attention [21]	0.713 (0.559 – 0.908)	0.562 (0.447 – 0.677)	100.2 (48.7 – 151.6)
	FCN	0.900 (0.891 – 0.908)	0.688 (0.676 – 0.701)	35.0 (28.7 – 41.2)
	RNN [20]	0.871 (0.840 – 0.901)	0.694 (0.653 – 0.735)	45.2 (33.2 – 57.1)
	DCNN [17]	0.911 (0.890 – 0.932)	0.710 (0.686 – 0.734)	31.6 (21.6 – 41.6)
	FCNN [18]	0.923 (0.900 – 0.946)	0.776 (0.748 – 0.804)	27.4 (17.7 – 37.1)
	XGBM [16]	0.905 (0.852 – 0.947)	0.730 (0.758 – 0.801)	38.0 (14.1 – 41.9)
	TVAE	**0.911 (0.866 – 0.956)**	**0.737 (0.659 – 0.814)**	**31.0 (15.1 – 46.8)**
16	BiLSTM + Attention [21]	0.776 (0.730 – 0.823)	0.475 (0.348 – 0.601)	73.0 (55.9 – 90.0)
	FCN	0.876 (0.862 – 0.893)	0.650 (0.615 – 0.685)	39.8 (35.2 – 44.4)
	RNN [20]	0.883 (0.855 – 0.910)	0.718 (0.689 – 0.733)	38.0 (29.8 – 46.1)
	DCNN [17]	0.899 (0.876 – 0.922)	0.708 (0.648 – 0.768)	32.8 (26.3 – 39.3)
	FCNN [18]	0.929 (0.908 – 0.950)	0.785 (0.734 – 0.836)	23.0 (17.4 – 28.6)
	XGBM [16]	0.933 (0.925 – 0.951)	0.719 (0.646 – 0.754)	25.2 (9.6 – 28.8)
	TVAE	**0.934 (0.901 – 0.966)**	**0.720 (0.652 – 0.787)**	**21.2 (10.9 – 31.4)**

worrisome problem in imbalanced datasets. Therefore, our proposed framework almost solved the imbalance problem.

4.3. VTSA

In VTSA, we perform stratified train/test split with the value of split size (test size) = s, $s \in [0.3, 0.4, 0.5]$. Given a predefined list of values for s, the training set is diminished in size, and concurrently, the test set is augmented until both sets achieve approximate parity in size ($s = 0.5$). This experiment validates the performance of the comparison models against the variability of research data. Experiment results of VTSA for CNUH and UV are shown in Table 9. In this experiment, we only use representative comparison methods that have good performance in the case of K-Fold CV, including RNN, DCNN, and XGBM, along with the proposed TVAE. For experimenting across multi-study cases, we choose the configuration of $D = 8$, and custom step size $k = 1$. This represents a generic scenario fraught with various challenges for prediction models.

In CNUH's results, with test size = 0.3, TVAE achieved 0.938 of AUROC, 0.874 of AUPRC, 36.0 of Late Alarm, 0.742 of Kappa, and 0.873 of F1 score. This result is only worse than XGBM across all comparison metrics. Nevertheless, as the test size was increased to 0.4, XGBM's performance notably declined, whereas TVAE experienced only a slight decrease, emerging as the model with the highest performance across all metrics. When test size = 0.5, TVAE continues to outperformance the comparison methods on all aspects. In UV's results, in all three cases of test size value, TVAE shows superior results compared to the comparison methods. The typical case with split size = 0.3, TVAE is higher than 0.073 of AUROC, 0.099 of AUPRC, 0.074 of Kappa, and 0.082 of F1 score, and reduces 629 of Late Alarm samples, in compared to XGBM.

We extract some test samples and represent their confusion matrices in Fig. 8 for CNUH, and Fig. 9 for UV. Overall, TVAE (Fig. 8(a)) demonstrates a well-balanced performance with a notable number of true positives (806 samples) and a relatively low count of false negatives

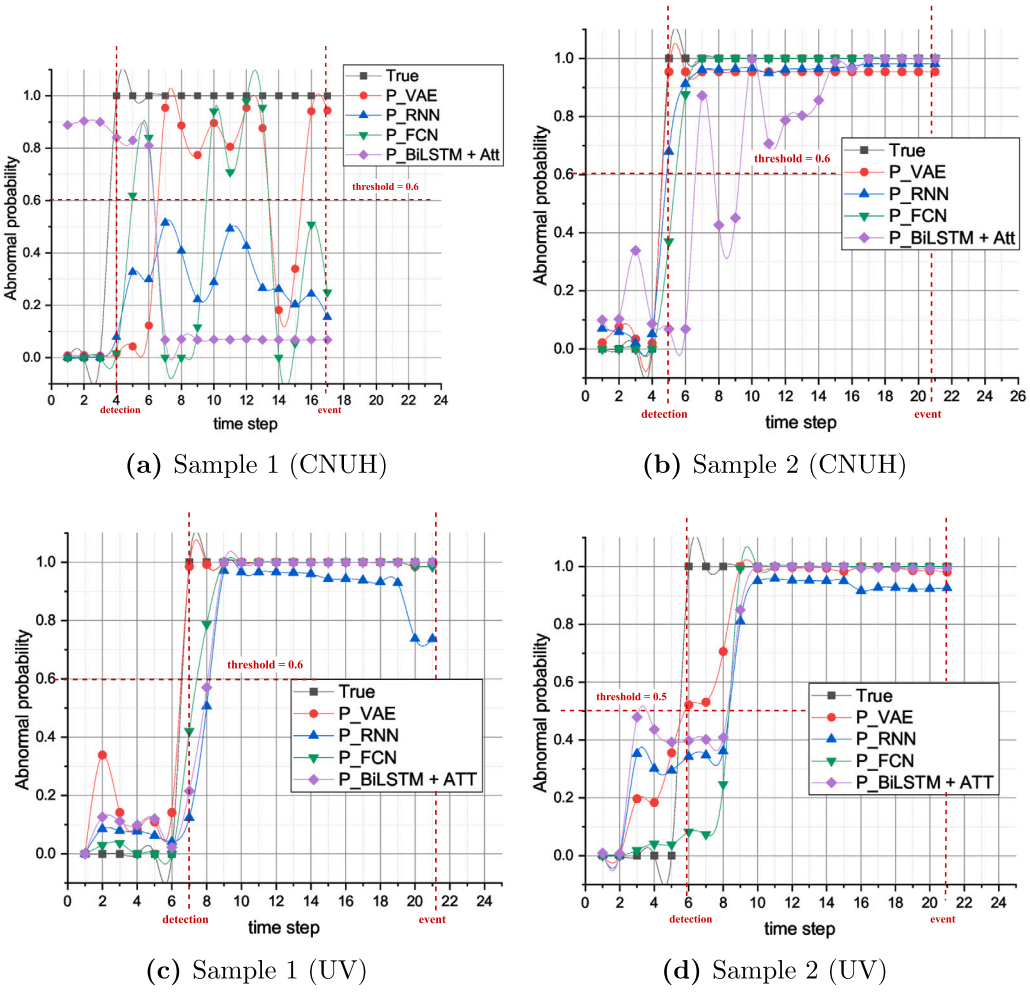


Fig. 4. Sample results with Window size = 8; predict next 1 h case. True, P_VAE, P_RNN, P_FCN, P_BiLSTM+Att indicate ground truth and event probability results returned by TVAE, RNN, FCN, and BiLSTM + Attention, respectively. In the 'event interval' (mentioned in Section 3), the higher the probability, the better the model performed. For the threshold selection, the choice of the threshold is based on the purpose of increasing the precision of all methods and minimizing the Late Alarm situation. In Fig. 4(a), the probability result of the proposed method (red curve) shows that TVAE can detect the most 'abnormal points' in the 'event interval' compared to other methods. Although there are still 'late alarm' points (the 6th, 14th time points, etc.), TVAE has most significantly limited this situation. In the samples at Fig. 4(b) (CNUH), Fig. 4(c) and Fig. 4(d) (UV), TVAE got perfect results with zero 'late alarm' points. TVAE achieves perfect results with zero 'late alarm' points and also avoids 'false alarm' cases where at points outside of 'event interval' the proposed method returns a low abnormal probability (below the threshold).

(28 samples). Notably, in the context of RRS, to have a system capable of triggering a patient's abnormality to the treatment team as soon as possible, special attention should be paid to the rate of type II error (late alarm or false negative) [51–53]. In this respect, RNN (Fig. 8(b)) performed well in detecting positive cases (781 samples) but had a significantly higher number of false negatives than TVAE with 42 late alerts samples.

Fig. 9 shows the confusion matrix of comparison approaches on UV with the setting of split size = 0.3. TVAE (Fig. 9(a)) had a notable number of correctly predicted event classes (1,183 samples) but was accompanied by a notable number of mispredicted positives (102 samples). However, the rate of type II error was the lowest (280 samples) among comparison methods. RNN (Fig. 9(b)) increases 54 late alarm samples for this type of error.

In this analysis, TVAE consistently outperformed RNN and other comparison methods in most cases, demonstrating strong performance across varied test sizes and prediction periods. Its stability and superior performance make it an appealing option for predictive modeling in this situation. The resilience of TVAE is validated in these results, showcasing its consistent performance across different data sizes compared to other methods like RNN and DCNN. While comparison methods may achieve robust performance in specific test size values, they exhibit substantial performance decline in other cases.

4.4. LOOCV

In this investigation, LOOCV was specifically designed to assess the model's performance under conditions where the challenge of data imbalance has been addressed, yet a limitation of a small data sample size. To simulate this problem, we perform data under-sampling to balance the distribution between two classes: event and non-event on CNUH. After the RUD process, the high imbalance situation at CNUH has disappeared with a balance in the number of samples between the two classes (339 samples for each class). Nevertheless, training and validating prediction models with a dataset of 678 samples presents a substantial challenge. LOOCV is applied as an efficient cross-validation method using a single data point that serves as the validation set, while the remaining data is used for training in each iteration.

4.4.1. Statistical significance of classification using a binomial cumulative distribution

After applying RUD and obtaining a restricted set of samples, all data preceding the model training stage must undergo WIP. The inherent 'truncate' characteristic of WIP further contributes to a reduction in the number of samples, influenced by the specified sizes of D and k. To evaluate model performance on the limited amount of available data, we extract chance-level classification results from LOOCV and compare

Table 7
Experiment results on UV dataset. Window size (D) = 8. Custom window step size k = [8; 10; 16]. Predict the next 8/10/16 hour's clinical status.

Window step size (k)	Model	Metric (95% CI)		
		AUROC	AUPRC	Late alarm
8	BiLSTM + Attention [21]	0.896 (0.858 – 0.934)	0.736 (0.684 – 0.789)	887.4 (551.8 – 1222.9)
	FCN	0.968 (0.959 – 0.967)	0.871 (0.855 – 0.886)	255.8 (185.1 – 326.4)
	RNN [20]	0.964 (0.962 – 0.967)	0.874 (0.870 – 0.877)	286.4 (260.6 – 312.2)
	DCNN [17]	0.925 (0.911 – 0.940)	0.777 (0.752 – 0.802)	631.6 (513.9 – 749.3)
	FCNN [18]	0.958 (0.942 – 0.973)	0.885 (0.862 – 0.909)	361.8 (227.5 – 496.1)
	XGBM [16]	0.901 (0.890 – 0.911)	0.778 (0.756 – 0.799)	863.4 (773.9 – 952.9)
	TVAE	**0.983 (0.973 – 0.993)**	**0.898 (0.877 – 0.919)**	**124.0 (34.6 – 213.3)**
10	BiLSTM + Attention [21]	0.903 (0.873 – 0934)	0.755 (0.703 – 0.806)	817.2 (547.3 – 1087.1)
	FCN	0.968 (0.963 – 0.972)	0.871 (0.858 – 0.884)	254.8 (213.1 – 296.5)
	RNN [20]	0.961 (0.954 – 0.967)	0.863 (0.851 – 0.875)	322.6 (263.1 – 382.1)
	DCNN [17]	0.917 (0.900 – 0.935)	0.764 (0.744 – 0.784)	696.6 (542.2 – 851.0)
	FCNN [18]	0.961 (0.951 – 0.970)	0.887 (0.877 – 0.898)	332.6 (250.1 – 415.1)
	XGBM [16]	0.900 (0.892 – 0.908)	0.776 (0.761 – 0.790)	861.2 (792.5 – 929.9)
	TVAE	**0.984 (0.981 – 0.986)**	**0.902 (0.892 – 0.911)**	**116.6 (91.4 – 141.7)**
16	BiLSTM + Attention [21]	0.890 (0.855 – 0.926)	0.718 (0.663 – 0.773)	896.6 (586.1 – 1207.1)
	FCN	0.960 (0.951 – 0.968)	0.837 (0.837 – 0.873)	314.4 (243.6 – 385.1)
	RNN [20]	0.958 (0.953 – 0.962)	0.857 (0.849 – 0.865)	335.4 (295.2 – 375.6)
	DCNN [17]	0.919 (0.905 – 0.933)	0.756 (0.734 – 0.778)	657.8 (541.5 – 774.1)
	FCNN [18]	0.960 (0.950 – 0.971)	0.879 (0.869 – 0.889)	323.2 (226.0 – 420.4)
	XGBM [16]	0.898 (0.892 – 0.903)	0.774 (0.763 – 0.786)	857.6 (814.8 – 900.4)
	TVAE	**0.980 (0.970 – 0.989)**	**0.894 (0.869 – 0.919)**	**145.4 (68.1 – 222.7)**

Table 8
Kappa and F1 scores on two dataset. Window size (D) = 8. Custom window step size k = [8; 10; 16]. Predict the next 8/10/16 hour's clinical status.

Dataset		CNUH		UV	
Window time size (D)	Model	Kappa	F1	Kappa	F1
8	BiLSTM + Attention [21]	0.592	0.714	0.705	0.850
	FCN	0.687	0.828	0.773	0.932
	RNN [20]	0.698	0.841	0.774	0.933
	DCNN [17]	0.697	0.840	0.728	0.877
	FCNN [18]	0.698	0.841	0.776	0.935
	XGBM [16]	0.698	0.841	0.718	0.866
	TVAE	**0.699**	**0.843**	**0.786**	**0.947**
16	BiLSTM + Attention [21]	0.453	0.546	0.714	0.861
	FCN	0.688	0.829	0.773	0.932
	RNN [20]	0.686	0.827	0.769	0.927
	DCNN [17]	0.698	0.841	0.720	0.868
	FCNN [18]	0.729	0.879	0.780	0.940
	XGBM [16]	0.694	0.837	0.723	0.872
	TVAE	**0.696**	**0.839**	**0.787**	**0.949**
24	BiLSTM + Attention [21]	0.554	0.668	0.696	0.839
	FCN	0.668	0.805	0.766	0.923
	RNN [20]	0.696	0.839	0.767	0.924
	DCNN [17]	0.696	0.839	0.717	0.864
	FCNN [18]	0.734	0.885	0.776	0.935
	XGBM [16]	0.781	0.941	0.722	0.870
	TVAE	**0.703**	**0.847**	**0.784**	**0.945**

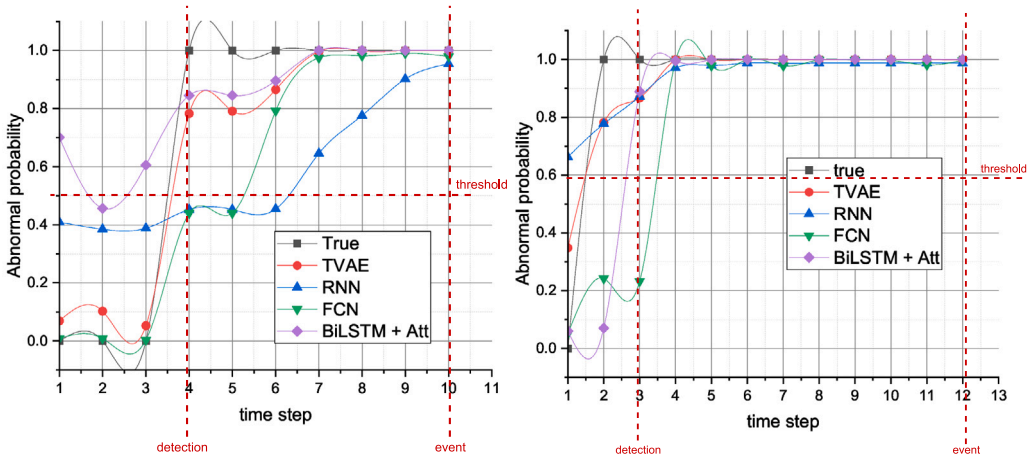
them with statistical significance thresholds for each variation of data size. For a given class c, the theoretical percentage chance grading level is given as $100/c$. In this study, with $c = 2$, the chance level is $100/2 = 50\%$. This threshold is established under the assumption of an infinite sample size. In our limited number of sample data, the empirical chance level is influenced. To address this problem, we can assess the statistical significance of decoding accuracy by considering classification errors through a binomial cumulative distribution [29].

Based on several studies on medical signal classification [29,54–56], we use binomial cumulative distribution to derive statistical significance thresholds $St(\alpha) = \text{binoinv}(1-\alpha, n, 1/c) \times 100/n$, where α is the significance level given by $\alpha = f/n$ (i.e. the ratio of tolerated false positives f — the number of observations correctly classified by chance — to all observations f). For instance, with $c = 2$ in our classification task, the threshold accuracy at $p = 0.05$ on 100 samples is 58%. It suggests that any model's performance below 70% is not statistically significant.

4.4.2. Chance-level variation results

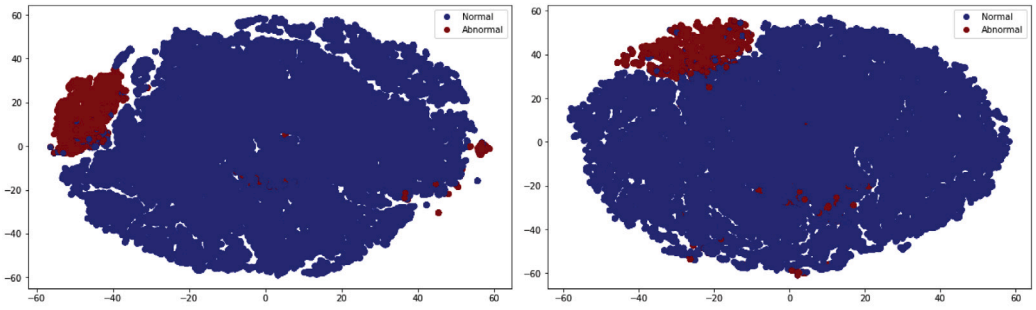
Fig. 10 presents the classification results of TVAE, RNN, and XGBM using accuracy (%) in LOOCV. In Fig. 10(a), except a minor decrease observed in sample sizes between 100 and 130, the accuracy percentages of the compared models exhibit a distinct pattern from the threshold accuracy across various p values. This observation underscores the efficacy of the models under the WIP treatment, demonstrating their ability to overcome limitations associated with the number of experimental samples. Notably, TVAE (depicted by the green curve) outperforms other models, maintaining an accuracy percentage fluctuating between 84% and 89%. While XGBM (dark yellow curve) initially achieves high performance (over 90%), it experiences a decline at around 100 to 130 samples, failing to match the accuracy of TVAE as the sample size increases.

Additional chance-level variation results are presented in Fig. 10(b), extending the prediction horizon to 16 h ($k = 16$). Despite fluctuations in threshold accuracies, the performance trends of the compared



(a) Sample 1 (CNUH) (b) Sample 2 (CNUH)

Fig. 5. Sample results with Window size = 8; predict next 8 h case in CNUH dataset. True, TVAE, RNN, FCN, BiLSTM+Att indicates ground truth and event probability results returned by TVAE, RNN, FCN, and BiLSTM + Attention, respectively. In both Fig. 5(a), TVAE (red curve) could detect all 'abnormal timepoint' between 'event interval' while FCN (green curve) and RNN (blue curve) still shows some missing detection in this interval (time point 4th, 5th and 6th). BiLSTM with Attention approach returns a similar result to our proposed method in this sample. However, at the time interval outside the 'event interval', false alarms can occur with this method (time point 1st, 3rd) because of its high sensitivity while TVAE still has stable performance. Similar results happened in Fig. 5(b), all methods demonstrate satisfactory performance in detecting events, while still yielding occasional false alarms during normal intervals.



(a) Proposed TVAE (b) RNN

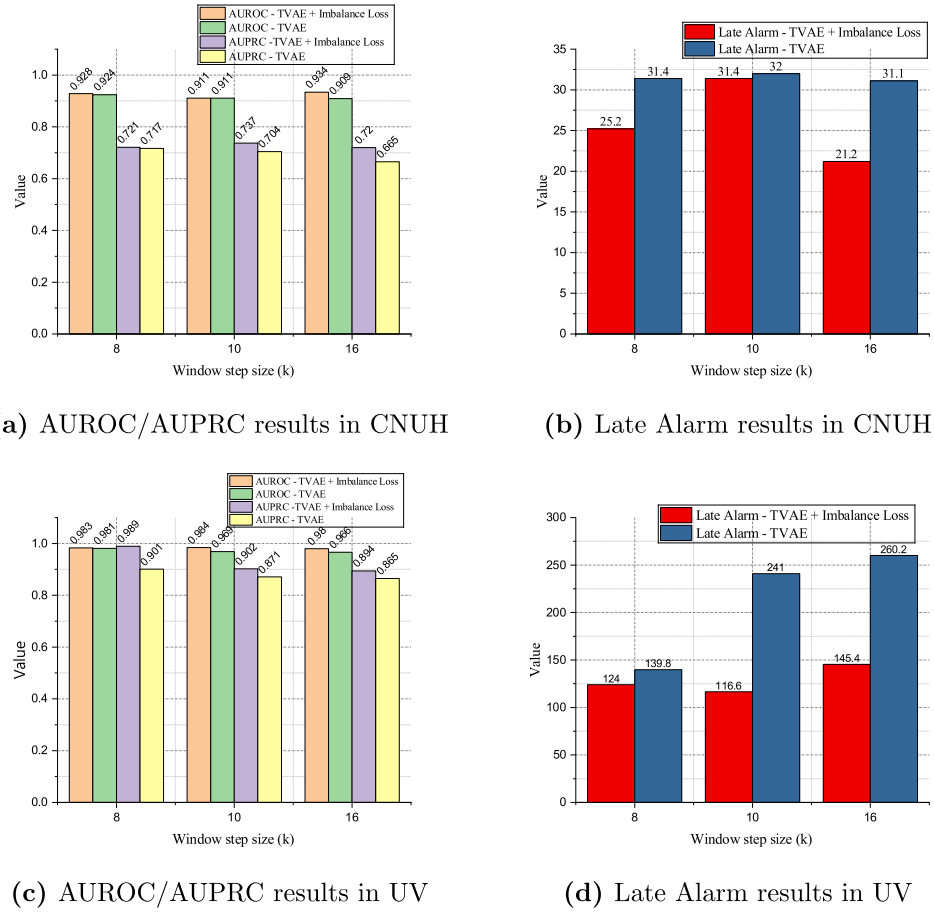
Fig. 6. Performance of feature space representation on comparison methods using t-SNE on UV dataset. The Brown area shows the distribution of the abnormal class; the blue shows the distribution of the normal class. Figs. 6(a) and 6(b) visualize the features space representation of TVAE and RNN, respectively. In Fig. 6(a), TVAE clearly separates two classes. In Fig. 6(b), it can be noted that the two classes have been divided into 2 regions but the borderline is conjoined and there is undefined separation indicating the poor separation ability of RNN. This result suggests that the TVAE has learned a discriminative feature representation of the data that is useful for classification and that this is likely due to the unsupervised nature of VAE training, which allows the model to learn more robust and generalizable features.

models diverge, notably experiencing a pronounced decrease in the sample size range from 30 to 70 samples. In detail, TVAE witnessed a 25% reduction in accuracy, RNN saw a 33% decline, and XGBM exhibited a 31% deterioration of accuracy. Subsequently, performance trends of the comparison approaches indicate recovery, with TVAE consistently maintaining a leading position, achieving accuracy levels ranging from 71% to 82%. Despite occasional degradation in specific cases, the overall performance of the methods appears to be resilient to variations in sample sizes.

A worth-notice aspect drawn from Fig. 10 is that, while the performance of the models exhibits resilience to changes in the number of sample sizes, it is notably affected by variations in the WIP configuration. Specifically, with a fixed window size $D = 8$, improving the step size k from 8 to 16 leads to a more pronounced decrease in model performance. This underscores the capacity of WIP to emulate realistic scenarios in the context of in-hospital clinical deterioration prediction, where forecasting outcomes in the more distant future becomes inherently more challenging, given the same historical data conditions. Moreover, our proposed system demonstrates significant progress in mitigating the challenges posed by limited sample sizes.

5. Conclusion

This study proposed an application of DL on early warning systems that used the advantage of LSTM and VAEs for learning features to represent the distribution of clinical information data in a temporal context and overcome the late alarm problem. Additionally, the class separation ability helps our system to make DEWS record high scores with high sensitivity compared with the related approaches. Our optimal multi-task pipeline learning overcomes the imbalance issue. It also improves the quality of the latent representation through reconstruction loss, focal loss, and Dice loss. It also prevents bias for DEWS. In addition, the window interval sliding mechanism increases the system's customizability according to the requirements of the problem. This study is the first to use these methods on RRS tasks. Therefore, our research shows a promising future in clinical settings and expands the experiment



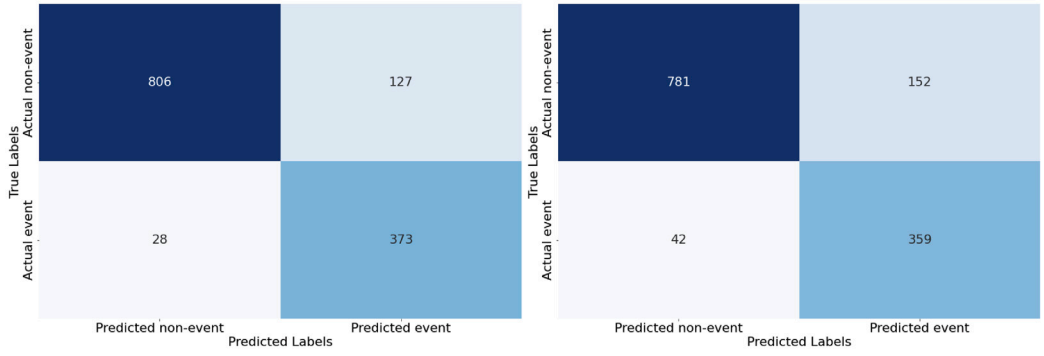
(a) AUROC/AUPRC results in CNUH

(b) Late Alarm results in CNUH

(c) AUROC/AUPRC results in UV

(d) Late Alarm results in UV

Fig. 7. The variation in the value of performance metrics of TVAE with and without Imbalance Loss in different future prediction intervals. Figs. 7(a) and 7(b) show the results of AUROC/AUPRC and Late Alarm in the CNUH dataset; Fig. 7(c) and Fig. 7(d) show the results of AUROC/AUPRC and Late Alarm in UV dataset, respectively. In Fig. 7(a), AUROC of TVAE + Imbalance Loss tends to increase despite the expansion of future prediction interval values, from 0.911 (next 10 h case) to 0.934 (next 16 h case), while without Imbalance Loss, TVAE tends to decrease in performance. As for the AUPRC value, the methods all tend to decrease. However, when using Imbalance Loss, our proposed method always achieves better results and there is an increase when extending the future prediction period, from 0.721 (next 8 h case) to 0.737 (next 10 h case) compared to when without Imbalance Loss. In the Late Alarm comparison, the lower values of Late Alarm, the better of method's performance. The use of Imbalance Loss significantly improved the results of TVAE when the Late Alarm value declined remarkably from 31.0 in the next 10 h case to 21.2 in the next 16 h case. The results in the UV dataset show more challenges as the comparison values tend to decrease as the future prediction period increases. However, when using Imbalance Loss, the proposed method shows better results on all metrics.



(a) TVAE

(b) RNN

Fig. 8. Confusion matrix of comparison models on CNUH with configuration of $D = 8$; $k = 1$; split size = 0.4.

Table 9
Experiment results of VTSA on two datasets with D = 8, k = 1. Variation of train/test split size s = [0.3, 0.4, 0.5]. Predict the next 1 hour's clinical status.

Dataset	Split size	Method	AUROC	AUPRC	Late alarm	Kappa	F1
CNUH	0.3	RNN [20]	0.904	0.772	39	0.721	0.870
		DCNN [17]	0.928	0.798	41	0.714	0.862
		XGBM [16]	0.952	0.943	28	0.781	0.942
		TVAE	**0.938**	**0.874**	**36**	**0.742**	**0.873**
	0.4	RNN [20]	0.903	0.759	42	0.679	0.787
		DCNN [17]	0.911	0.741	48	0.652	0.759
		XGBM [16]	0.927	0.825	32	0.682	0.822
		TVAE	**0.935**	**0.829**	**28**	**0.741**	**0.827**
	0.5	RNN [20]	0.884	0.743	93	0.613	0.761
		DCNN [17]	0.899	0.762	98	0.616	0.765
		XGBM [16]	0.903	0.811	79	0.685	0.806
		TVAE	**0.907**	**0.813**	**61**	**0.723**	**0.871**
UV	0.3	RNN [20]	0.944	0.822	334	0.747	0.849
		DCNN [17]	0.915	0.878	583	0.699	0.802
		XGBM [16]	0.902	0.789	972	0.674	0.779
		TVAE	**0.973**	**0.888**	**280**	**0.759**	**0.861**
	0.4	RNN [20]	0.938	0.811	562	0.723	0.816
		DCNN [17]	0.923	0.804	717	0.694	0.802
		XGBM [16]	0.898	0.791	841	0.682	0.781
		TVAE	**0.951**	**0.824**	**465**	**0.765**	**0.822**
	0.5	RNN [20]	0.881	0.762	1465	0.641	0.716
		DCNN [17]	0.854	0.745	1914	0.496	0.692
		XGBM [16]	0.84	0.736	2374	0.465	0.681
		TVAE	**0.905**	**0.815**	**959**	**0.698**	**0.754**

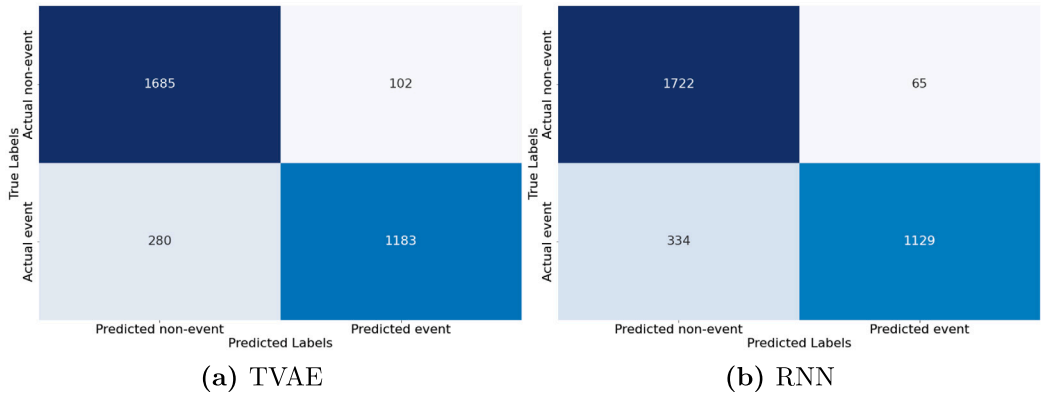


Fig. 9. Confusion matrix of comparison models on UV with configuration of $D = 8$; $k = 1$; split size = 0.3.

for different configurations. With support from doctors, the proposed system is under the integration process with real-time hospital systems to improve the quality of clinical care. The system is expected to create a breakthrough step for the application of DL in clinical prediction and decision support.

CRediT authorship contribution statement

Trong-Nghia Nguyen: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Project administration, Methodology, Data curation, Conceptualization. **Soo-Hyung Kim:** Writing – review & editing, Supervision, Methodology, Funding acquisition, Conceptualization. **Bo-Gun Kho:** Validation, Supervision, Resources, Data curation, Conceptualization. **Nhu-Tai Do:** Investigation, Formal analysis, Conceptualization. **Ngumimi-Karen Iyortsuun:** Writing – original draft, Formal analysis. **Guee-Sang Lee:** Supervision, Resources, Formal analysis. **Hyung-Jeong Yang:** Supervision, Resources, Funding acquisition, Data curation.

Declaration of competing interest

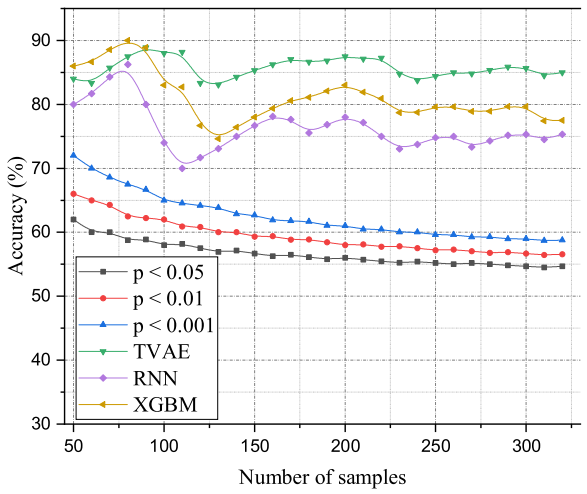
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

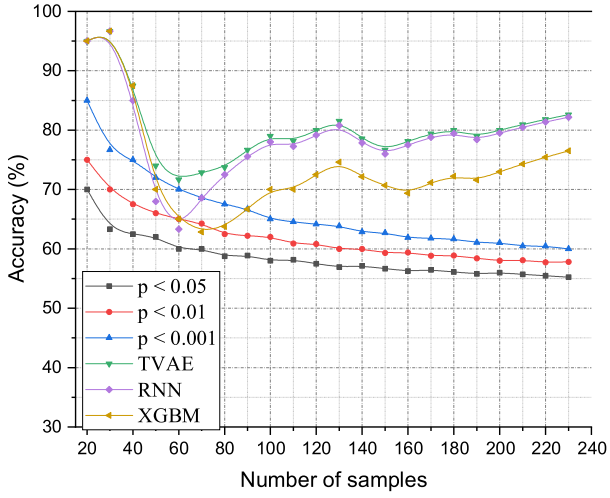
The authors do not have permission to share data.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government(MSIT) (RS-2023-00219107), and Institute of Information & communications Technology Planning & Evaluation (IITP) under the Artificial Intelligence Convergence Innovation Human Resources Development (IITP-2023-RS-2023-00256629) grant funded by the Korea government(MSIT). This study was supported by a grant (HCRI23001) of the Chonnam National University Hwasun hospital Institute for Biomedical Science.



(a) Chance-level results in $k = 8$



(b) Chance-level results in $k = 16$

Fig. 10. Chance-level results using accuracy (%) of LOOCV in CNUH from across two study cases: $k = 8$ (Fig. 10(a)) and $k = 16$ (Fig. 10(b)). The green, dark yellow and violet curves show the accuracy results of TVAE, XGBM, and RNN, respectively. Blue, red, and black curves illustrate threshold accuracy corresponding to each p value = 0.05, 0.01, 0.001.

References

[1] J.-m. Kwon, Y. Lee, Y. Lee, S. Lee, H. Park, J. Park, Validation of deep-learning-based triage and acuity score using a large national dataset, PLoS One 13 (10) (2018) e0205836, http://dx.doi.org/10.1371/journal.pone.0205836.

[2] P.S. Chan, B.K. Nallamothu, H.M. Krumholz, L.H. Curtis, Y. Li, B.G. Hammill, J.A. Spertus, Readmission rates and long-term hospital costs among survivors of an in-hospital cardiac arrest, Circul. Cardiovas. Qual. Outcomes 7 (6) (2014) 889–895, http://dx.doi.org/10.1161/CIRCOUTCOMES.114.000925.

[3] T.D. Gunter, N.P. Terry, The emergence of national electronic health record architectures in the United States and Australia: models, costs, and questions, J. Med. Internet Res. 7 (1) (2005) e383, http://dx.doi.org/10.2196/jmir.7.1.e3.

[4] D.A. Jones, M.A. DeVita, R. Bellomo, Rapid-response teams, New Engl. J. Med. 365 (2) (2011) 139–146, http://dx.doi.org/10.1056/NEJMra0910926.

[5] P.G. Lyons, D.P. Edelson, M.M. Churpek, Rapid response systems, Resuscitation 128 (2018) 191–197, http://dx.doi.org/10.1016/j.resuscitation.2018.05.013.

[6] A.H. Taenzer, B.C. Spence, The afferent limb of rapid response systems: continuous monitoring on general care units, Crit. Care Clin. 34 (2) (2018) 189–198, http://dx.doi.org/10.1016/j.ccc.2017.12.001.

[7] B.G. Candel, R. Duijzer, M.I. Gaakeer, E. Ter Avest, Ö. Sir, H. Lameijer, R. Hessels, R. Reijnen, E.W. van Zwet, E. de Jonge, et al., The association between vital signs and clinical outcomes in emergency department patients of different age categories, Emerg. Med. J. 39 (12) (2022) 903–911, http://dx.doi.org/10.1136/emermed-2020-210628.

[8] Emergency Cardiovascular Care, Part 4: Systems of Care and Continuous Quality Improvement (18), S397–S413, http://dx.doi.org/10.1161/CIR.0000000000000258.

[9] B.Y. Lee, S.-B. Hong, Rapid response systems in Korea, Acute Crit. Care 34 (2) (2019) 108–116, http://dx.doi.org/10.4266/acc.2019.00535.

[10] Y.J. Lee, J.J. Park, Y.E. Yoon, J.W. Kim, J.S. Park, T. Kim, J.H. Lee, J.-W. Suh, Y.H. Jo, S. Park, et al., Successful implementation of a rapid response system in the department of internal medicine, Korean J. Critical Care Med. 29 (2) (2014) 77–82, http://dx.doi.org/10.4266/kjccm.2014.29.2.77.

[11] G.B. Smith, D.R. Prytherch, P.E. Schmidt, P.I. Featherstone, Review and performance evaluation of aggregate weighted 'track and trigger'systems, Resuscitation 77 (2) (2008) 170–179, http://dx.doi.org/10.1016/j.resuscitation.2007.12.004.

[12] C.P. Subbe, M. Kruger, P. Rutherford, L. Gemmel, Validation of a modified Early Warning Score in medical admissions, Qjm 94 (10) (2001) 521–526, http://dx.doi.org/10.1093/qjmed/94.10.521.

[13] W. Weng, X. Zhu, INet: convolutional networks for biomedical image segmentation, IEEE Access 9 (2021) 16591–16603, http://dx.doi.org/10.1109/ACCESS.2021.3053408.

[14] C.J. Bae, S. Griffith, Y. Fan, C. Dunphy, N. Thompson, J. Urchek, A. Parchman, I.L. Katzan, The challenges of data quality evaluation in a joint data warehouse, eGEMs 3 (1) (2015) http://dx.doi.org/10.13063/2327-9214.1125.

[15] Y. Hirano, Y. Kondo, K. Sueyoshi, K. Okamoto, H. Tanaka, Early outcome prediction for out-of-hospital cardiac arrest with initial shockable rhythm using machine learning models, Resuscitation 158 (2021) 49–56, http://dx.doi.org/10.1016/j.resuscitation.2020.11.020.

[16] J. Liu, J. Wu, S. Liu, M. Li, K. Hu, K. Li, Predicting mortality of patients with acute kidney injury in the ICU using XGBoost model, Plos one 16 (2) (2021) e0246306, http://dx.doi.org/10.1371/journal.pone.0246306.

[17] W. Caicedo-Torres, J. Gutierrez, ISeeU: Visually interpretable deep learning for mortality prediction inside the ICU, J. Biomed. Inform. 98 (2019) 103269, http://dx.doi.org/10.1016/j.jbi.2019.103269.

[18] S. Le, A. Allen, J. Calvert, P.M. Palevsky, G. Braden, S. Patel, E. Pellegrini, A. Green-Saxena, J. Hoffman, R. Das, Convolutional neural network model for intensive care unit acute kidney injury prediction, Kidney Int. Rep. 6 (5) (2021) 1289–1298, http://dx.doi.org/10.1016/j.ekir.2021.02.031.

[19] P. Chen, W. Dong, J. Wang, X. Lu, U. Kaymak, Z. Huang, Interpretable clinical prediction via attention-based neural network, BMC Med. Inform. Decis. Mak. 20 (3) (2020) 1–9, http://dx.doi.org/10.1186/s12911-020-1110-7.

[20] J.-m. Kwon, Y. Lee, Y. Lee, S. Lee, J. Park, An algorithm based on deep learning for predicting in-hospital cardiac arrest, J. Am. Heart Assoc. 7 (13) (2018) e008678, http://dx.doi.org/10.1161/JAHA.118.008678.

[21] F.E. Shamout, T. Zhu, P. Sharma, P.J. Watkinson, D.A. Clifton, Deep interpretable early warning system for the detection of clinical deterioration, IEEE J. Biomed. Health Inform. 24 (2) (2019) 437–446, http://dx.doi.org/10.1109/JBHI.2019.2937803.

[22] S.M. Lauritsen, M. Kristensen, M.V. Olsen, M.S. Larsen, K.M. Lauritsen, M.J. Jø rgensen, J. Lange, B. Thiesson, Explainable artificial intelligence model to predict acute critical illness from electronic health records, Nature Commun. 11 (1) (2020) 3852, http://dx.doi.org/10.1038/s41467-020-17431-x.

[23] A. Tversky, D.J. Koehler, Support theory: A nonextensional representation of subjective probability, Psychol. Rev. 101 (4) (1994) 547, http://dx.doi.org/10.1016/0165-4896(94)90008-6.

[24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Adv. Neural Inf. Process. Syst. 30 (2017) http://dx.doi.org/10.48550/arXiv.1706.03762.

[25] G.B. Smith, D.R. Prytherch, P. Meredith, P.E. Schmidt, P.I. Featherstone, The ability of the National Early Warning Score (NEWS) to discriminate patients at risk of early cardiac arrest, unanticipated intensive care unit admission, and death, Resuscitation 84 (4) (2013) 465–470, http://dx.doi.org/10.1016/j.resuscitation.2012.12.016.

[26] C. Lea, R. Vidal, A. Reiter, G.D. Hager, Temporal convolutional networks: A unified approach to action segmentation, in: Computer Vision–ECCV 2016 Workshops: Amsterdam, the Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III 14, Springer, 2016, pp. 47–54, http://dx.doi.org/10.1007/978-3-319-49409-8_7.

[27] M. Swapna, U.M. Viswanadhula, R. Aluvalu, V. Vardharajan, K. Kotecha, Bio-signals in medical applications and challenges using artificial intelligence, J. Sensor Actuator Netw. 11 (1) (2022) 17, http://dx.doi.org/10.3390/jsan11010017.

[28] M.M. Baig, H. GholamHosseini, A.A. Moqeem, F. Mirza, M. Lindén, A systematic review of wearable patient monitoring systems–current challenges and opportunities for clinical adoption, J. Med. Syst. 41 (2017) 1–9, http://dx.doi.org/10.1007/s10916-017-0760-1.

[29] E. Combrisson, K. Jerbi, Exceeding chance level by chance: The caveat of theoretical chance levels in brain signal classification and statistical assessment of decoding accuracy, J. Neurosci. Methods 250 (2015) 126–136, http://dx.doi.org/10.1016/j.jneumeth.2015.01.010.

[30] A. Vabalas, E. Gowen, E. Poliakoff, A.J. Casson, Machine learning algorithm validation with a limited sample size, PLoS One 14 (11) (2019) e0224365, http://dx.doi.org/10.1371/journal.pone.0224365.

[31] T.J. Moss, M.T. Clark, J.F. Calland, K.B. Enfield, J.D. Voss, D.E. Lake, J.R. Moorman, Cardiorespiratory dynamics measured from continuous ECG monitoring improves detection of deterioration in acute care patients: A retrospective cohort study, PLoS One 12 (8) (2017) e0181448, http://dx.doi.org/10.1371/journal.pone.0181448.

[32] L. Sörnmo, P. Laguna, Electrocardiogram (ECG) signal processing, in: Wiley Encyclopedia of Biomedical Engineering, John Wiley & Sons, Inc., Hoboken, NJ, USA, 2006, http://dx.doi.org/10.1002/9780471740360.ebs1482.

[33] D.P. Kingma, M. Welling, Auto-encoding variational bayes, 2013, http://dx.doi.org/10.48550/arXiv.1312.6114, arXiv preprint arXiv:1312.6114.

[34] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780, http://dx.doi.org/10.1162/neco.1997.9.8.1735.

[35] M. Vafaeipour, O. Rahbari, M.A. Rosen, F. Fazelpour, P. Ansarirad, Application of sliding window technique for prediction of wind velocity time series, Int. J. Energy Environ. Eng. 5 (2014) 1–7, http://dx.doi.org/10.1007/s40095-014-0105-5.

[36] J.-S. Chou, N.-T. Ngo, Time series analytics using sliding window metaheuristic optimization-based machine learning system for identifying building energy consumption patterns, Appl. Energy 177 (2016) 751–770, http://dx.doi.org/10.1016/j.apenergy.2016.05.074.

[37] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (7553) (2015) 436–444, http://dx.doi.org/10.1038/nature14539.

[38] X. Li, X. Sun, Y. Meng, J. Liang, F. Wu, J. Li, Dice loss for data-imbalanced NLP tasks, 2019, http://dx.doi.org/10.18653/v1/2020.acl-main.45, arXiv preprint arXiv:1911.02855.

[39] A. Spangher, J. May, S.-r. Shiang, L. Deng, Multitask learning for class-imbalanced discourse classification, 2021, http://dx.doi.org/10.48550/arXiv.2101.00389, arXiv preprint arXiv:2101.00389.

[40] K. Phinzi, D. Abriha, S. Szabó, Classification efficacy using k-fold cross-validation and bootstrapping resampling techniques on the example of mapping complex gully systems, Remote Sens. 13 (15) (2021) 2980, http://dx.doi.org/10.3390/rs13152980.

[41] L. Van der Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (11) (2008).

[42] M.S. Wisz, R. Hijmans, J. Li, A.T. Peterson, C. Graham, A. Guisan, NCEAS Predicting Species Distributions Working Group, Effects of sample size on the performance of species distribution models, Divers. Distrib. 14 (5) (2008) 763–773, http://dx.doi.org/10.1016/S0304-3800(01)00388-X.

[43] L. Budach, M. Feuerpfeil, N. Ihde, A. Nathansen, N. Noack, H. Patzlaff, F. Naumann, H. Harmouch, The effects of data quality on machine learning performance, 2022, http://dx.doi.org/10.48550/arXiv.2207.14529, arXiv preprint arXiv:2207.14529.

[44] M.A. Tahir, J. Kittler, F. Yan, Inverse random under sampling for class imbalance problem and its application to multi-label classification, Pattern Recognit. 45 (10) (2012) 3738–3750, http://dx.doi.org/10.1016/j.patcog.2012.03.014.

[45] T. O'Malley, E. Bursztein, J. Long, F. Chollet, H. Jin, L. Invernizzi, et al., KerasTuner, 2019, https://github.com/keras-team/keras-tuner.

[46] J. Bergstra, Y. Bengio, Random search for hyper-parameter optimization, J. Mach. Learn. Res. 13 (10) (2012) 281–305, URL http://jmlr.org/papers/v13/bergstra12a.html.

[47] B. Shahriari, K. Swersky, Z. Wang, R.P. Adams, N. De Freitas, Taking the human out of the loop: A review of Bayesian optimization, Proc. IEEE 104 (1) (2015) 148–175, http://dx.doi.org/10.1287/educ.2018.0188.

[48] S. Abenna, M. Nahid, H. Bouyghf, B. Ouacha, EEG-based BCI: A novel improvement for EEG signals classification based on real-time preprocessing, Comput. Biol. Med. 148 (2022) 105931, http://dx.doi.org/10.1016/j.compbiomed.2022.105931.

[49] S. Abenna, M. Nahid, H. Bouyghf, B. Ouacha, An enhanced motor imagery EEG signals prediction system in real-time based on delta rhythm, Biomed. Signal Process. Control 79 (2023) 104210, http://dx.doi.org/10.1016/j.bspc.2022.104210.

[50] H.R. Sofaer, J.A. Hoeting, C.S. Jarnevich, The area under the precision-recall curve as a performance metric for rare binary events, Methods Ecol. Evolut. 10 (4) (2019) 565–577, http://dx.doi.org/10.1111/2041-210X.13140.

[51] J.A. Freiman, T.C. Chalmers, H.A. Smith, R.R. Kuebler, The importance of beta, the type II error, and sample size in the design and interpretation of the randomized controlled trial: survey of two sets of "negative" trials, in: Medical Uses of Statistics, CRC Press, 2019, pp. 357–389, http://dx.doi.org/10.1056/NEJM197809282991304.

[52] C. Ueki, G. Sakaguchi, Importance of awareness of Type II Error, Ann. Thoracic Surg. 105 (1) (2018) 333, http://dx.doi.org/10.1016/j.athoracsur.2017.03.062.

[53] H. Matsuda, The importance of type II error and falsifiability, Human Ecol. Risk Assess. Int. J. 11 (1) (2005) 189–200, http://dx.doi.org/10.1080/10807030590920015.

[54] K.K. Ang, C. Guan, K.S.G. Chua, B.T. Ang, C. Kuah, C. Wang, K.S. Phua, Z.Y. Chin, H. Zhang, Clinical study of neurorehabilitation in stroke using EEG-based motor imagery brain-computer interface with robotic feedback, in: 2010 Annual International Conference of the IEEE Engineering in Medicine and Biology, IEEE, 2010, pp. 5549–5552, http://dx.doi.org/10.1109/IEMBS.2010.5626782.

[55] E. Demandt, C. Mehring, K. Vogt, A. Schulze-Bonhage, A. Aertsen, T. Ball, Reaching movement onset-and end-related characteristics of EEG spectral power modulations, Front. Neurosci. 6 (2012) 65, http://dx.doi.org/10.3389/fnins.2012.00065.

[56] T. Pistohl, A. Schulze-Bonhage, A. Aertsen, C. Mehring, T. Ball, Decoding natural grasp types from human ECoG, Neuroimage 59 (1) (2012) 248–260, http://dx.doi.org/10.1016/j.neuroimage.2011.06.084.