A Deep Learning Approach for Inpatient Length of Stay and Mortality Prediction

Junde Chen[a], Trudi Di Qi[a], Jacqueline Vu[a], Yuxin Wen[a]*

[a] Fowler School of Engineering, Chapman University, Orange, 92866, CA, USA
* Corresponding author, e-mail: yuwen@chapman.edu

Abstract

Purpose: Accurate prediction of the Length of Stay (LoS) and mortality in the Intensive Care Unit (ICU) is crucial for effective hospital management, and it can assist clinicians for real-time demand capacity (RTDC) administration, thereby improving healthcare quality and service levels.

Methods: This paper proposes a novel one-dimensional (1D) multi-scale convolutional neural network architecture, namely 1D-MSNet, to predict inpatients' LoS and mortality in ICU. First, a 1D multi-scale convolution framework is proposed to enlarge the convolutional receptive fields and enhance the richness of the convolutional features. Following the convolutional layers, an atrous causal spatial pyramid pooling (SPP) module is incorporated into the networks to extract high-level features. The optimized Focal Loss (FL) function is combined with the synthetic minority over-sampling technique (SMOTE) to mitigate the imbalanced-class issue.

Results: On the MIMIC-IV v1.0 benchmark dataset, the proposed approach achieves the optimum *R-Square* and *RMSE* values of 0.57 and 3.61 for the LoS prediction, and the highest test accuracy of 97.73% for the mortality prediction.

Conclusion: The proposed approach presents a superior performance in comparison with other state-of-the-art, and it can effectively perform the LoS and mortality prediction tasks.

Keywords: SMOTE; Multi-scale convolution; Atrous causal SPP; Length of stay prediction; Mortality prediction.

1. Introduction

Offering timely patient care while sustaining high resource utilization is a key challenge that most hospitals currently and continuously face. In the United States, each hour of delay in the patient transfer is related to an adjusted 3% increase in the odds of inpatient mortality [1]. From the perspective of healthcare providers, the economic pressure to deliver effective and accessible care has reached an unprecedented level [2]. Especially, the COVID-19 pandemic exposes and exacerbates the challenge to healthcare systems, which have been reporting substantial challenges in assuring and expanding their facilities' capacity to treat patients in the past two years [3], [4]. Patients' Length of Stay (LoS) in a hospital is a crucial indicator for the quality of care and effective allocation of healthcare resources [5], [6], [7]. Even with significant investments in critical care drugs, hospital healthcare resources are usually inadequate to meet the needs of inpatients, especially in developing countries. Meanwhile,

allocating healthcare resources to patients at high risk of death is also critical to patients' survival. Hospitals are under pressure to improve care efficiency and minimize costs. The two best practices, including real-time demand capacity (RTDC) management [8] and multidisciplinary discharge-focused rounds [9], have shown great promise. For example, RTDC predicts which and how many patients will be discharged daily. Utilizing daily demand projections, the hospital can prioritize the current patients who are ready for discharge. Apparently, RTDC management relies on accurate prediction of individual patient discharges. Thereupon, there is an urgent need and realistic importance to develop new systems that can automatically predict the LoS and mortality risk for inpatients.

The prediction of LoS is a regression task that can be performed by analyzing various physical examination index data of patients. Even with its crucial importance, the LoS prediction for inpatients has received less attention than the mortality prediction in the literature owing to the difficulty of this task. In recent research, data-driven and machine learning (ML) methods have been employed for healthcare data analysis, including risk prediction with electronic health records [10], mining signatures from event sequences [11], and predicting patients' LoS in hospitals [12], [15], [16], etc. Peres et al. [12] developed a structured data-driven model to predict the individual LoS and the risk of prolonged Intensive Care Unit (ICU) stay on a large ICU dataset from different Brazilian hospitals. Despite the impressive performance obtained, their ensemble learning (EL) method integrates up to 8 ML models. Integrating and managing multiple baseline models within the EL framework can increase the complexity and memory requirements of the system. Moreover, EL models can be prone to overfitting and underfitting if the aggregation method is too simple or too complex, or if the base models are too strong or too weak. This can lead to overestimating or underestimating the uncertainty and variability of the data. Therefore, if the baseline models are not properly selected and individually overfit the data, ensemble learning models may amplify the errors [13], [14]. Turgeman et al. [15] built a regression tree model called Cubist to predict the hospital LoS at the time of admission. However, the calculation process of their method is complicated since it maps all the LoS cases to higher dimensional spaces. Also, there is still room for improvement in the accuracy of the model. In another research, Alsinglawi et al. [16] introduced a new regressor architecture named stacking regressor to predict LoS in patients diagnosed with heart failure from electronic medical records. Their experimental results showed that their proposed stacking regressor outperformed other methods, such as deep learning-based regressors in their study. Although their proposed stacking regressor is a promising EL framework, the results of the final regressor strongly depend on the initial conditions. The final regressor is sensitive to the quality and diversity of the base models and the data, as it relies on the limitations and assumptions of the individual models, and on the representativeness and independence of the data samples and features. More recently, popular deep learning (DL) methods, such as convolutional neural networks (CNN) and temporal convolution networks (TCN), have been proposed for LoS prediction tasks and achieved positive prediction efficacy [17], [18]. Fu et al. [17] developed four DL models, and they confirmed that the CNN outperformed other comparative candidates. Zhou et al. [19] recommended a CNN

architecture primarily comprised of fully connected (FC) and convolutional layers to predict the LoS in heart failure patients. CNNs require a fixed input size and may lose details in the pooling layer of the network [20], [21]. Rocheteau et al. [18] proposed an improved TCN, temporal pointwise convolution network (TPC), for predicting LoS in the ICU. TPC combines temporal convolutional layers with pointwise convolutions to extract temporal and inter-feature information. However, the TPC has a high model size and may require more data storage. Moreover, the interrelationship between different variables is prone to be ignored [46], [47]. Multi-scale convolution kernels with different convolutional receptive fields have proven effective for many tasks [22], [23], [24], which is not considered in recent research such as [17], [18], [25].

On another front, mortality prediction belongs to a classification problem, and ML-based methods have also been employed to establish classification models in complex medical settings, such as the ICU [26], [27], [28], [29]. Bao et al. [30] trained seven ML algorithms and estimated their performance in predicting patients' mortality or survival during hospitalization. Compared with the traditional unregularized logistic regression method, Ruzicka et al. [31] built an XGBoost ML model to predict patients' mortality in hospitals. Ganapathy et al. [32] developed seven ML models, namely, Artificial neural network (ANN), Naïve Bayes (NB), Logistic regression (LR), Bayesian regression (BLR), Binary Discriminant analysis (BDA), K-nearest neighbor (KNN), and Random Forest to predict the mortality in patients diagnosed with oral squamous cell carcinoma. Likewise, using three ML methods, including the ANN, support vector machine (SVM), and Random Forest, Lin et al. [28] predicted the in-hospital mortality for ICU patients with acute kidney injury. Despite reasonably good findings reported in the literature, the traditional ML methods also suffer from some bottlenecks, such as reliance on hand-designed features, lack of robustness, risk of overfitting, and low accuracy. DL technique has also been utilized for the mortality prediction task in recent decades. Caicedo-Torres et al. [33] trained a CNN model, namely ISeeU2, to perform the mortality prediction of patients inside the ICU, and they used the MIMIC-III dataset for evaluating models. Using the TF-IDF representation of nurse notes as input, Khine et al. [34] introduced an ensemble CNN model to predict 30-day ICU mortality on the MIMIC-III benchmark dataset. Based on computerized tomography (CT) image data, Roopa et al. [35] employed several well-known architectural models like Resnet18, Resnet50, and Resnet101 to perform the mortality prediction of lung cancer and their experimental results revealed that the CNN classifier is better than SVM. Moreover, CNN-based DL methods have also proven to be efficient in the references [36], [37], [38]. These research findings show the significance of associating DL methods with clinical diagnosis and health care. However, the methods stated above ignore the serious class-imbalance problem in patients' mortality or survival, which can result in low accuracy of mortality prediction. Besides, as mentioned previously, multi-scale feature representation and dilated receptive fields that prevent information leakage during modeling processes are not considered in most previous research. Therefore, to fill this research gap, we propose a novel one-dimensional CNN architecture, namely 1D-MSNet, to predict patients' in-hospital LoS and mortality in the ICU. The pre-processing is

first implemented for the original sample data. By using the SMOTE, we enlarge the sample size of the mortality class so that it is close to the number of samples in the survival class. Then, in the bottom convolutional layers of the network, the 1D multi-scale convolution kernel is substituted for the single convolution kernel to expand the convolutional receptive fields and enhance the richness of the convolutional features. Following the bottom convolutional layers, the dilated causal SPP module, which includes 4 parallel 1D average pooling layers concatenated by 4 parallel 1D dilated causal convolutional layers and up-sampling layers, is incorporated into the network for extracting high-level data features. Besides that, the traditional Focal Loss (FL) function is improved to replace the Cross-Entropy (CE) loss function for alleviating the class-imbalance problem of samples in mortality prediction. To sum up, the major contributions of this work can be recapitulated below.

- A novel one-dimensional convolution network architecture, which we termed 1D-MSNet, is proposed to predict patients' in-hospital LoS and mortality in the ICU. Especially for the mortality class, the SMOTE is utilized to synthesize new sample data for alleviating the class imbalance problem.
- The 1D multi-scale convolution kernel is substituted for the single convolution kernel in the bottom convolutional layers of the network to expand the receptive fields and enhance the richness of convolutional features.
- The dilated causal SPP module is embedded into the networks for extracting high-level data features. The traditional FL function is optimized to replace the CE loss function to eliminate the class-imbalance influence in mortality prediction.

The remainder of this paper is structured as follows: Section 2 primarily discusses the methodology to predict patients' in-hospital LoS and mortality in the ICU. Section 3 is devoted to the algorithm experiments, and a series of experiments are conducted along with comparative analysis. Section 4 concludes this paper with a summary and recommendations for future work.

Statement of Significance

Problem or Issue	Patient Length of stay (LoS) and mortality are main indicators for the timely patient care and healthcare resource utilization. LoS and mortality can be affected by a multitude of different factors and can vary based on different patients' conditions. The question is how to accurately predict LoS and mortality based on patients medical records.
What is Already Know	Although there have been many data driven and machine learning approaches available, most research treat both tasks as classification problems, also, most deep learning approaches failed to prevent information leakage during modeling processes.
What this Paper Adds	This study proposes a novel one-dimensional CNN architecture, namely 1D-MSNet, the 1D multi-scale convolution kernel is substituted for the single convolution kernel in the bottom convolutional layers of the network to expand the receptive fields and enhance the richness of convolutional features. A dilated causal causal spatial pyramid pooling (SPP) module is embedded into the networks for extracting high-level data features. The proposed model can be generalized for other clinical applications.

2. Methodology

2.1 Synthetic minority oversampling technique

For the sample imbalance problem, the random under-sampling (RUS) method increases the number of samples in the minority category by randomly duplicating samples. However, it can easily result in overfitting. By improving the RUS algorithm, Chawla et al. [39] recommended the synthetic minority oversampling technique (SMOTE) to synthesize new samples in minority categories. SMOTE applies an iterative search and selection approach. To synthesize new samples in minority categories, the amount of generated samples *N* and the threshold value of samples among *k*-nearest neighbors needs to be determined. Among them, the *N* refers to the sample size in the majority category, while *k* depends upon the number of samples to be synthesized. This process continues until the desired number of samples in the minority category is generated. Concretely, the specific steps of sample data balancing for mortality prediction are presented as follows. First, mortality sample data in the original samples are taken as the learning set for the synthetic samples. The Euclidean distances of in-hospital mortality samples are calculated to measure the similarity and enable newly generated samples closest to the original data. The formulas are presented as follows.

$$d(x,s) = \sqrt{\sum_{i=1}^{n}(x_i - s_i)^2} \tag{1}$$

$$S_{new} = S + (X - S) * rand(0,1) \tag{2}$$

Among them, $d(x,s)$ denotes the Euclidean distance between any two samples. S_{new} implies the newly generated mortality sample data, *S* is the original in-hospital mortality (minority class) samples that are to be oversampled, *X* is one of the nearest neighbors of the original mortality sample, $*$ symbolizes dot products of elements, and rand(0, 1) implies an arbitrary number in the interval [0, 1]. Then, a sample is randomly chosen from the inpatient mortality class, and *k*-nearest neighbor samples are chosen according to the computed Euclidean distance. Referring to the literature [40], the *k* value is assigned as 5 here. Iteratively, the above steps are repeated *N* times for synthesizing *N* new inpatient mortality samples. Algorithm 1 depicts the specific procedure of the SMOTE-based mortality sample balance.

Algorithm 1: *SMOTE*(*k, N, T*) based sample balance.

Input: The number of samples in the minority category *T,* the amount of *SMOTE N*%,
and the number of *k*-nearest neighbors (Here *k*=5).

Output: *T* * (*N*/100) synthetic samples in the minority category

If *N* < 100

then, randomize *T* samples in the minority category:

T = *T* * (*N*/100)

N = 100

end

S[]: array for the initial samples in the minority category

S_{new}[]: array for the synthetic samples

$X[]$: K neighboring samples of the array of original minority category samples
For $i \leftarrow 1$ *to* T
  Choose an arbitrary number between 1 and k, which is called j
    $dif = X[j] - S[i]$
    gap = arbitrary number in the interval [0, 1]
    $S_{new}[i] = S[i] + gap * dif$
end
Return S_{new}

2.1. Atrous causal SPP

In a typical convolutional network architecture, the pooling layer often follows one or multiple convolutional layers to scale down the spatial size of feature maps while decreasing the number of training parameters and computing costs. Nevertheless, the pooling operation also makes the network lose the information needed to detect precise relationships among object parts. To alleviate this problem, spatial pyramid pooling [20], abbreviated as SPP, which maps local features to different dimensional spaces and fuses them, is introduced to aggregate convolution features and gain more feature representations with rich spatial information. For an SPP with the grid size of $M \times M$, the size of the convolution kernel depicted by $k=k_h=k_w$ can be calculated by $k_h=[h/M]$ and $k_w=[w/M]$, where w and h represent the width and height of an intermediate feature map, and $[\cdot]$ indicates the ceiling operation. Fig. 1 depicts the structure of the classical SPP module. Note that, the classical SPP is designed for image processing in the computer vision (CV) field. In this work, however, the tabular data are used in our analysis instead of the image data. Therefore, the classical SPP structure is modified, and the original two-dimensional (2D) convolution and pooling layers are replaced by the 1D ones. Besides, the atrous causal convolution is substituted for the regular convolution in the SPP module to capture long-term temporal dependencies while maintaining computational efficiency, considering that convolution kernels cannot be as large as expected with the increase of parameters and computing costs. In particular, the regular convolution kernel has the disadvantage that the spatial resolution of feature maps is halved at each step. For this reason, the 1D atrous causal SPP module is designed in our network, and the hyperparameter of the dilated rate is set to $r=2$. Compared with the traditional SPP, this optimized 1D atrous causal SPP, namely ACSPP, increases the convolutional receptive field while reducing the computational overhead.

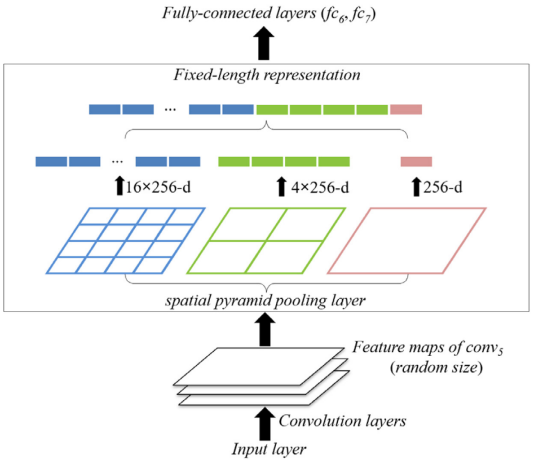


Fig. 1. The network structure of classical SPP [20].

2.3. Proposed approach

2.3.1 1D-MSNet architecture

Motivated by the previous study [41], we propose a 1D multi-scale atrous causal SPP convolutional network, which we termed 1D-MSNet, to predict the inpatients' LoS and mortality in the ICU using the patients' clinical records. Fig. 2 demonstrates the overall architecture of the proposed 1D-MSNet. As portrayed in Fig. 2, the network is primarily composed of two crucial modules, including the backbone multi-scale convolutional network and an enhanced 1D atrous causal SPP module. Due to the limited receptive field of the single-scale convolution kernel, the extracted information is relatively unitary, which limits the performance of model prediction to a certain extent. Therefore, in this study, the multi-scale convolution kernel is substituted for the single convolution kernel in the bottom convolutional layers of the 1D-MSNet, to enlarge the convolutional receptive fields and enhance the richness of the convolutional features. The multi-scale convolution operations are conducted on intermediate feature maps with the convolution kernel (filter) sizes of 1, 3, and 5, respectively. More specifically, following 2 consecutive convolutional layers, a Batch Normalization (BN), Rectified Linear Unit (ReLU), and 1D max-pooling (MAP) layers are repeated 4 times. These operations reduce the internal covariance shift and computational complexity of the model. Consequently, the backbone network of the proposed 1D-MSNet comprises a total of eight convolutional layers, and the number of convolution kernels is 32, 64, 128, and 64 with the sizes of 3, 5, 1, 5, 3, 5, 1, and 5, respectively. After that, connecting the bottom multi-scale convolutional layers, the dilated causal SPP module, which includes 4 parallel 1D average pooling layers followed by 4 parallel 1D atrous causal convolutional layers and up-sampling layers, is embedded into the network for extracting high-level data features. It is worth noting that the atrous convolution has a larger receptive field than the standard convolution and is more suitable for handling long-distance dependencies. The causal convolution is a strictly time-constrained operation that is dedicated to addressing sequence data and avoiding series information leakage from the future to the past. Concretely, the specific descriptions of the atrous and causal operations are presented below.

(1) Atrous convolution. For a given input sequence $\boldsymbol{s} \in \mathbb{R}^n$ and filter f: $\{0, \ldots, k\text{-}1\} \rightarrow \mathbb{R}$, the atrous

convolution operation F on the sequence element x of the sequence can be written as

$$F(x) = \sum_{i=0}^{k-1} f(i)\boldsymbol{s}_{x-d\cdot i} \tag{3}$$

where d represents the atrous factor, k is the filter size. $x - d \cdot i$ allows the network to see longer temporal features for the same number of convolution layers.

(2) Causal convolution. The role of causal convolution can be abstracted to predict y_n for a given sequence $s = \{x_0, x_1, ..., x_n\}$ and corresponding filter $F = \{f_0, f_1, ..., f_K\}$. The formula of causal convolution can be expressed as

$$y_n = \sum_{k=0}^{K} f_k x_{n-K+k} \tag{4}$$

In this study, the parallel 1D atrous causal convolutional layers are designed to extract discriminative features, the average pooling layers are responsible for scaling down the spatial size of the feature map which in turn reduces the parameter number and computing costs, and the up-sampling layers aim to increase the resolution of features so that the network can better discriminate among complicated data. At last, a concatenated layer is used to aggregate the features output from the parallel layers, and a flatten layer followed by a dense layer or completely linked (CL) Softmax layer is used for the regression or classification tasks, respectively. Table 1 summarizes the major parameters of the proposed 1D-MSNet architecture.

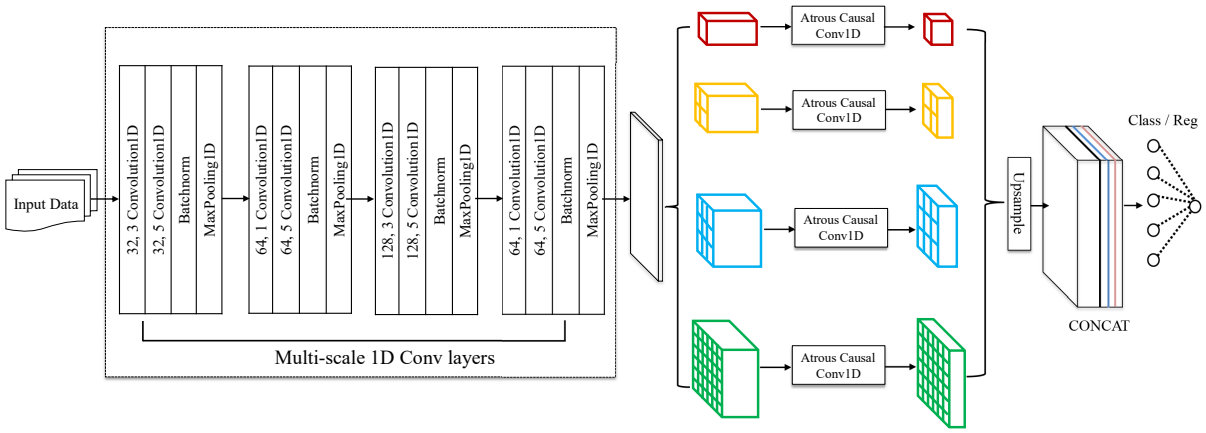


Fig. 2. The architecture of the proposed 1D-MSNet.

Table 1. The major parameters of 1D-MSNet.

Layer (module)	Input shape	Filter number	Kernel size	Output shape	Param number	Repeated
Input layer	(None, 52, 1)	-	-	(None, 32, 1)	-	1
Multi-scale Conv1d	(None, 52, 1)	32	3, 5	(None, 52, 32)	128+5152	2
BatchNormalization	(None, 52, 32)	-	-	(None, 52, 32)	128	1
MaxPooling1d	(None, 52, 32)	-	-	(None, 26, 32)	-	1
Multi-scale Conv1d	(None, 26, 32)	64	1, 5	(None, 26, 64)	2112+20544	2
BatchNormalization	(None,26, 64)	-	-	(None, 26, 64)	256	1
MaxPooling1d	(None, 26, 64)	-	-	(None, 13, 64)	-	1
Multi-scale Conv1d	(None,13,64)	128	3, 5	(None, 13, 128)	24704+82048	2
BatchNormalization	(None,13,128)	-	-	(None, 13, 128)	512	1

MaxPooling1d	(None, 13, 64)	-	-	(None, 7, 64)	-	1
Multi-scale Conv1d	(None,7, 128)	64	1, 5	(None, 7, 64)	8256+20544	2
BatchNormalization	(None, 7, 64)	-	-	(None, 7, 64)	256	1
AvgPooling1d	(None, 7, 64)	-	-	(None, 3, 64)	-	4
Atrous Causal Conv1d	(None, 3, 64)	-	1, *r*=2	(None, 3, 64)	4160×4	4
Upsampling	(None, 3, 64)	-	-	(None, 6, 64)	-	4
Concatenate	(None, 6, 64)	-	-	(None, 6, 256)	-	1
Flatten	(None, 6, 256)	-	-	(None, 1536)	-	1
Softmax/Dense	(None, 1536)	-	-	2, 1	-	1

2.3.2 Loss function

(1) Loss function of LoS prediction. In general, LoS prediction is a regression problem, and the mean squared error (MSE) function is utilized as the loss function in our network for the LoS prediction task. The formula of the MSE loss function can be written by

$$L_{pred} = -\frac{\sum_{k=1}^{n}(y_k - y_k^{'})^2}{n} \tag{5}$$

where y_k denotes the supervised data, i.e., true value, $y_k^{'}$ represents the output of the networks (predicted value), and n indicates the number of samples.

(2) Loss function of mortality prediction. By comparison, the in-hospital mortality prediction belongs to a classification task, and the Cross-Entropy (CE) loss function is the mostly used loss function in deep learning models, as expressed by

$$L_{CE} = -\sum_{c=1}^{C} \ell_c \, log \, p(c|x) \tag{6}$$

In Eq. (6), c indexes the number of categories, $p(c|x)$ denotes the predicted probability of a specific sample x belonging to category c, and ℓ_c is an indicator variable (If c is equal to the actual category number of the sample, then $\ell_c = 1$; otherwise $\ell_c = 0$). However, the CE loss function treats the negative and positive samples as the same weight and its performance is suboptimal when dealing with imbalance class samples. For this reason, reference [42] recommends a Focal-Loss (FL) function which assigns different weights for the positive and negative samples, as expressed below.

$$L_{FL} = -\omega_c (1 - p(c|x))^{\delta} \, log(\, p(c|x)) \tag{7}$$

In Eq. (7), ω_c and δ are the hyper-parameters of the weighting factor and modulating factor, respectively. The traditional FL function is designed to handle binary issues in the field of object detection. On the ground of this, we further modified the FL function and made it suitable for multi-classification problems. The modified FL (MFL) function is utilized in our network for substituting the existing CE loss function, and the formulas of the MFL function are expressed as follows.

$$L_{MFL} = -\sum_{c=1}^{C} \omega_c (1 - p(c|x))^{\delta} \ell_c \, log(\, p(c|x)) \tag{8}$$

$$\omega_c = count(x)/count(x \in c) \tag{9}$$

$$\ell_c = \begin{cases} 1, c = actual\_category \\ 0, c \neq actual\_category \end{cases} \quad (10)$$

3. Experimental results and analysis

3.1 Experimental setup

We have conducted extensive experiments to verify the validity of the proposed approach. The algorithms are implemented using the software of Anaconda (Python 3.6), where the widely-used libraries, including Scikit-Learn, Keras, and Tensorflow are used and accelerated by GPU. The algorithms are conducted on the publicly-available datasets, such as the MIMIC-IV v1.0, COVID-19, and Kaggle benchmark datasets, to perform the LoS and mortality prediction tasks. The hardware configuration for using the Python DL framework to implement the LoS and mortality prediction includes AMD EPYC 7502P 32-Core Processor, 32 GB RAM, and RTX A6000 GPU, which are used for algorithm operation.

3.2 Experiments on MIMIC-IV v1.0 dataset

The Medical Information Mart for Intensive Care (MIMIC)-IV v1.0 dataset, which was released on June 22, 2022, focuses on expanding the data elements available for patients within MIMIC-III. Additional data sources, such as electronic medicine administration records, are incorporated in MIMIC-IV. It establishes a modular organization of constituent data, linking datasets with external departments and different data modalities. Multivariate critical care data are included in the MIMIC-IV dataset from the ICU stays for over 40,000 inpatients admitted from 2008 to 2019 [43]. All data records were deidentified, i.e., patient identifiers are removed according to the Health Insurance Portability and Accountability Act (HIPAA) Safe Harbor provision. Following the cohort selection procedure introduced by [18], we extracted a set of feature variables from the MIMIC-IV dataset to predict the LoS and mortality for the inpatients. Specifically, a total of 52 characteristic metrics, including blood, circulatory, endocrine, infectious, injury, mental, and others, are extracted from the following tables: ADMISSIONS, DIAGNOSIS_ICD, PATIENTS, and ICDSTAYS. These tables record the patients' clinical diagnosis and physical examination data in hospitals, and they are linked through the subject_id. Among them, the ADMISSIONS table provides information regarding patients' admission to the hospital. The information available contains the admission and discharge time, demographic data, the source of the admission, and so on. The DIAGNOSIS_ICD table includes the crucial indicator ICD_CODE, which is divided into 17 sub-categories in our work referring to the ICD9 scheme issued by the U.S. National Center for Health Statistics. The PATIENTS table provides the inpatients' age and gender information, and the ICDSTAYS table gives the Intensive Care Unit (ICU) information for each hospital admission. In this paper, we focus on two events: LoS and mortality. LOS is defined as the date and time between admission and discharge from the hospital, which is calculated by the difference between admission time and discharge time from the ADMISSIONS table. For the LoS prediction, patients who died at the hospital are dropped from the data as they may bias the LoS since they were

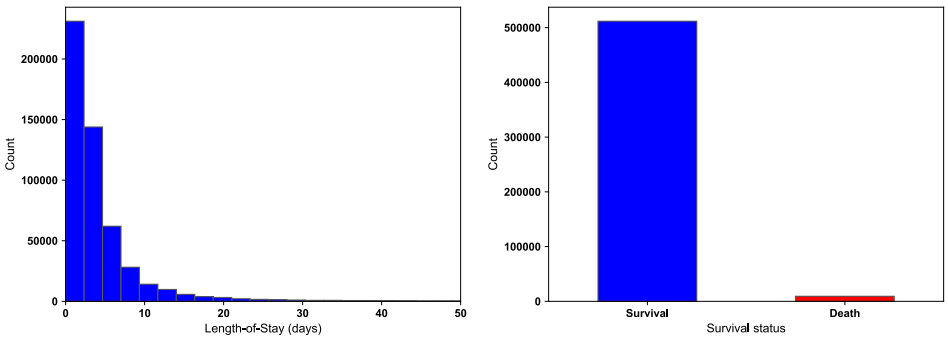
never discharged alive from the hospital. Whilst mortality is depicted by the hospital_expired_flag indicator in the ADMISSIONS table, which indicates the patients' death or survival in hospitals. Referring to the literature [52], we implemented the cohort inclusion and exclusion work for the original data. Except for the extracted useful indicators mentioned above, the unused columns like LANGUAGE, ADMIT_MIN, ADMISSION_LOCATION, and DOB were dropped from the original data. We also removed unnecessary columns and verified that there were no missing values in the dataset. Additionally, the negative indicator values, such as negative LoS caused by entry form error, were removed as they might skew prediction results. Furthermore, the extracted categorical variables, such as admission type, insurance type, age, ethnicity, religion, and marital status columns, were converted into dummy/indicator variables through One-Hot encoding. Table 2 displays the representative sample data extracted from the MIMIC-IV v1.0 database, and Table 3 summarizes the overall statistical description of the dataset. Fig. 3 presents the distribution of inpatients' LoS and mortality samples.

Table 2 The extracted sample data from the MIMIC-IV v1.0 database.

Subjectid	LoS	Mortality	Blood	Circulatory	Endocrine	...	Injury	Gender	Nervous	Pregnancy	Prenatal	Respiratory
18153382	1.61458	0	1	1	0	...	1	1	0	0	0	2
16378267	1.74166	0	0	5	1	...	1	1	0	0	0	0
16378267	8.13611	0	0	0	0	...	20	1	0	0	0	0
16378267	4.34722	1	0	0	0	...	27	1	0	0	0	0
18109892	8.20763	0	0	7	3		2	0	2	0	1	1
13104913	2.80138	0	0	0	0		1	1	0	4	0	0
13104913	3.57847	0	0	0	0	...	1	1	0	2	0	0
15813228	6.0625	0	1	4	3	...	5	0	0	0	0	0
18252284	2.59444	0	0	0	0	...	12	1	0	0	0	0
19984875	8.92638	0	0	0	0	...	17	1	0	0	0	0

Table 3 Overall statistical description for the dataset.

Variables (*n*)	*Statistic*	*Variables* (*n*)	*Statistic*	*Variables* (*n*)	*Statistic*
DECEASED = 1 (%)	9213 (1.8)	Neuro-SICU = 1 (%)	1807 (0.3)	GENDER = 1 (%)	269655 (51.8)
blood (mean (SD))	0.17 (0.48)	Neuro-Intermediate = 1 (%)	1770 (0.3)	INS-Medicaid = 1(%)	50717 (9.7)
circulatory (mean (SD))	0.94 (1.73)	Neuro-Stepdown = 1 (%)	830 (0.2)	INS-Medicare = 1(%)	170417 (32.7)
congenital (mean (SD))	0.03 (0.19)	ICU Type		INS-Other = 1(%)	299809(57.6)
digestive (mean (SD))	0.38 (0.88)	CVICU = 1 (%)	11464 (2.2)	Race	
endocrine (mean (SD))	0.67 (1.17)	CCU = 1 (%)	8330 (1.6)	ETH-asian = 1 (%)	24447 (4.7)
genitourinary(mean(S	0.28 (0.69)	MICU = 1 (%)	15732 (3.0)	ETH-black = 1 (%)	80137 (15.4)
infectious (mean (SD))	0.13 (0.44)	MICU-SICU = 1 (%)	12528 (2.4)	ETH-other = 1 (%)	50770 (9.7)
injury (mean (SD))	5.64 (6.81)	SICU = 1 (%)	11019 (2.1)	ETH-latino = 1 (%)	29740 (5.7)
mental (mean (SD))	0.38 (0.87)	TSICU = 1 (%)	8552 (1.6)	ETH-white = 1 (%)	335849 (64.5)
misc (mean (SD))	0.33 (0.75)	adm-ambulatory-observation	7118 (1.4)	AGE	
muscular (mean (SD))	0.22 (0.60)	adm-direct-emer =1(%)	21436(4.1)	AGE-newborn = 1 (%)	503817 (96.7)
neoplasms (mean (SD))	0.14 (0.53)	adm-direct-observation =1(%)	19691(3.8)	AGE-senior = 1 (%)	17126 (3.3)
nervous (mean (SD))	0.27 (0.67)	adm-elective =1(%)	71928 (13.8)	Marriage	
pregnancy (mean (SD))	0.09 (0.63)	adm-eu-observation =1(%)	100336(19.3)	MAR-divorced = 1	33525 (6.4)
prenatal (mean (SD))	0.15 (0.66)	adm-ew-emer =1(%)	157628(30.3)	MAR-married = 1 (%)	193794 (37.2)
respiratory (mean (SD))	0.21 (0.59)	adm-observation-admit =1(%)	55274 (10.6)	MAR-single = 1 (%)	179256 (34.4)
skin (mean (SD))	0.09 (0.39)	adm-surgical-same-day-admis	40078 (7.7)	MAR-unknown = 1	65761 (12.6)
Neuro type (%)	4407 (0.80)	adm-urgent =1(%)	47454 (9.1)	MAR-widowed = 1	48607 (9.3)



(a) Distribution of inpatients' LoS (b) Distribution of mortality

Fig. 3. The distribution of LoS and mortality.

It can be observed from Fig. 3 that most LoS are under 10 days, and the mortality sample distribution is extremely unbalanced. As seen in Fig. 3(b), the samples of the survival category (class 0) account for the majority, while the samples of the death category (class 1) account for only a very small number. Specifically, in a total of 520,943 rows of records, 9,213 rows are counted as mortality (see Table 3 for the statistic level description). Thereupon, the sample distribution is extremely unbalanced, and the number of deaths accounts for only a very small proportion. To alleviate the imbalance sample problem, the SMOTE algorithm is first utilized to augment the minority class samples in the training dataset to ensure a balanced distribution of positive and negative samples in the mortality prediction task.

We first perform the mortality prediction for patients in hospitals using the MIMIC-IV dataset. The dataset is split into training and test with a ratio of 3:1. Further, 30% of samples are drawn from the training dataset as the validation dataset. State-of-the-art methods, including the Multilayer Perceptron (MLP) classifier, KNN, SVM, Random Forest (RF), XGBoost, light gradient boosting machine (lightGBM), Temporal Convolutional Neural Network (TCNN), Transformer, and two-dimensional convolutional neural networks (2D-CNN) are chosen for comparison study. More than that, as mentioned in Section 2.1, uniformly random sampling from the interval [0, 1] is applied in SMOTE for data augmentation. Exploring alternative probability distribution functions, we replace the original uniform distribution in SMOTE with the half-normal function. Subsequently, we apply the 1D-MSNet for predicting inpatient mortality. We define this approach as 1D-MSNet (halfnorm). Besides, to explore the potential for performance enhancement through the integration of attention-based mechanisms, we incorporate a one-dimensional SEblock [44] into the networks, which is termed the 1D-MSNet (SE), for the comparison study on mortality prediction as well. It is worth mentioning that to ensure a fair comparison, the hyper-parameters of the compared models are kept consistent. For example, the adaptive moment estimation (Adam) [45] is utilized as a training optimizer of the network models, with a learning rate of 1×10^{-3}, 128 mini-batch size, and 30 epochs of training. The hyper-parameter n_estimators, which determines the number of trees in the models, is uniformly set to 70 for the tree algorithms such as the RF, XGBoost, and lightGBM, with the class number of two, automatic

tree_method, and the true silent. The commonly-used indicators including *Accuracy* (*Acc*), *Recall* (*Rec*), *F1-Score* (*F1*), *ROC-AUC*, and *PR-AUC* are utilized to investigate the performance of mortality prediction models. Figs. 4 and 5 depict the training performance of the proposed approach and the tested confusion matrices of different methods.

Table 4 The mortality prediction results of different methods.

Methods	Training set (%)			Validation set (%)			Time (s)
	Acc↑	*Rec*↑	*F1*↑	*Acc*↑	*Rec*↑	*F1*↑	
MLP	98.55±0.23	98.55±0.23	98.55±0.23	98.22±0.37	98.22±0.37	98.22±0.37	0:13:55
SVM	97.50±0.02	97.50±0.02	97.49±0.01	97.45±0.03	97.45±0.03	97.45±0.03	3:20:12
KNN	98.24±1.72	98.24±1.72	98.24±1.72	97.46±1.28	97.46±1.28	97.46±1.28	3:06:50
RF	99.29±0.04	99.29±0.04	99.29±0.04	99.01±0.06	99.01±0.06	99.01±0.06	0:04:38
XGBoost	98.79±0.03	98.79±0.03	98.79±0.03	98.59±0.02	98.59±0.02	98.59±0.02	0:04:05
LightGBM	95.56±0.13	95.56±0.13	95.56±0.13	95.49±0.14	95.49±0.14	95.49±0.13	0:02:36
TCNN	99.02±0.01	99.02±0.01	99.02±0.01	98.65±0.03	98.65±0.03	97.51±1.17	0:17:23
Transformer	97.15±1.42	97.16±1.46	97.15±1.42	97.11±1.00	97.09±0.93	97.11±1.00	0:57:06
2D-CNN	98.05±1.02	98.05±1.02	98.05±1.00	97.91±0.98	97.91±0.98	97.91±0.98	0:36:26
1D-MSNet(halfnorm)	99.49±0.42	99.49±0.42	99.49±0.42	98.94±0.05	98.94±0.05	98.94±0.05	0:27:14
1D-MSNet(SE)	99.26±0.24	99.26±0.24	99.26±0.24	98.73±0.23	98.73±0.23	98.73±0.23	0:33:04
1D-MSNet	99.53±0.05	99.53±0.05	99.53±0.05	98.91±0.01	98.91±0.01	98.91±0.01	0:24:00

↑ indicates the higher is better, while ↓ is in the reverse.

Table 5 The mortality prediction results on the test dataset.

Methods	Test set (%)				
	Acc↑	*Rec*↑	*F1*↑	*ROC-AUC*↑	*PR-AUC*↑
MLP	96.42±1.08	96.42±1.08	96.89±0.60	63.42±1.82	28.17±2.56
SVM	95.49±0.01	95.49±0.01	96.16±0.01	59.64±0.16	16.01±0.22
KNN	93.60±2.41	93.60±2.41	95.25±1.25	61.37±0.06	19.13±0.12
RF	97.58±0.36	97.58±0.36	97.44±0.05	56.29±0.25	23.15±0.14
XGBoost	97.76±0.01	97.76±0.01	97.64±0.01	63.94±0.25	31.37±0.47
LightGBM	90.60±0.26	90.60±0.26	93.41±0.14	54.22±0.05	8.91±0.06
TCNN	97.60±0.06	97.60±0.06	97.51±0.04	63.45±0.21	29.12±0.81
Transformer	93.26±1.25	93.26±1.25	94.82±1.65	55.65±5.65	10.15±6.51
2D-CNN	96.40±1.31	96.40±1.31	96.74±0.85	63.27±0.41	28.48±2.14
1D-MSNet(halfnorm)	97.82±0.11	97.82±0.11	97.65±0.06	63.05±0.63	31.13±0.51
1D-MSNet(SE)	97.95±0.09	97.95±0.09	97.69±0.02	61.64±2.19	31.21±1.63
1D-MSNet	97.73±0.12	97.73±0.12	97.65±0.02	64.19±1.01	32.41±0.71

↑ indicates the higher is better, while ↓ is in the reverse.

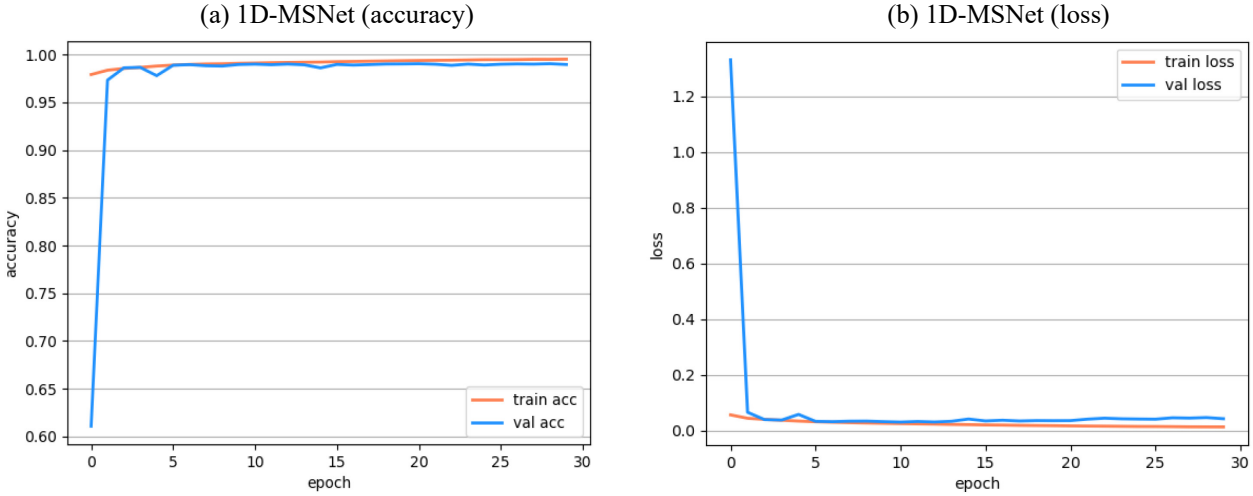
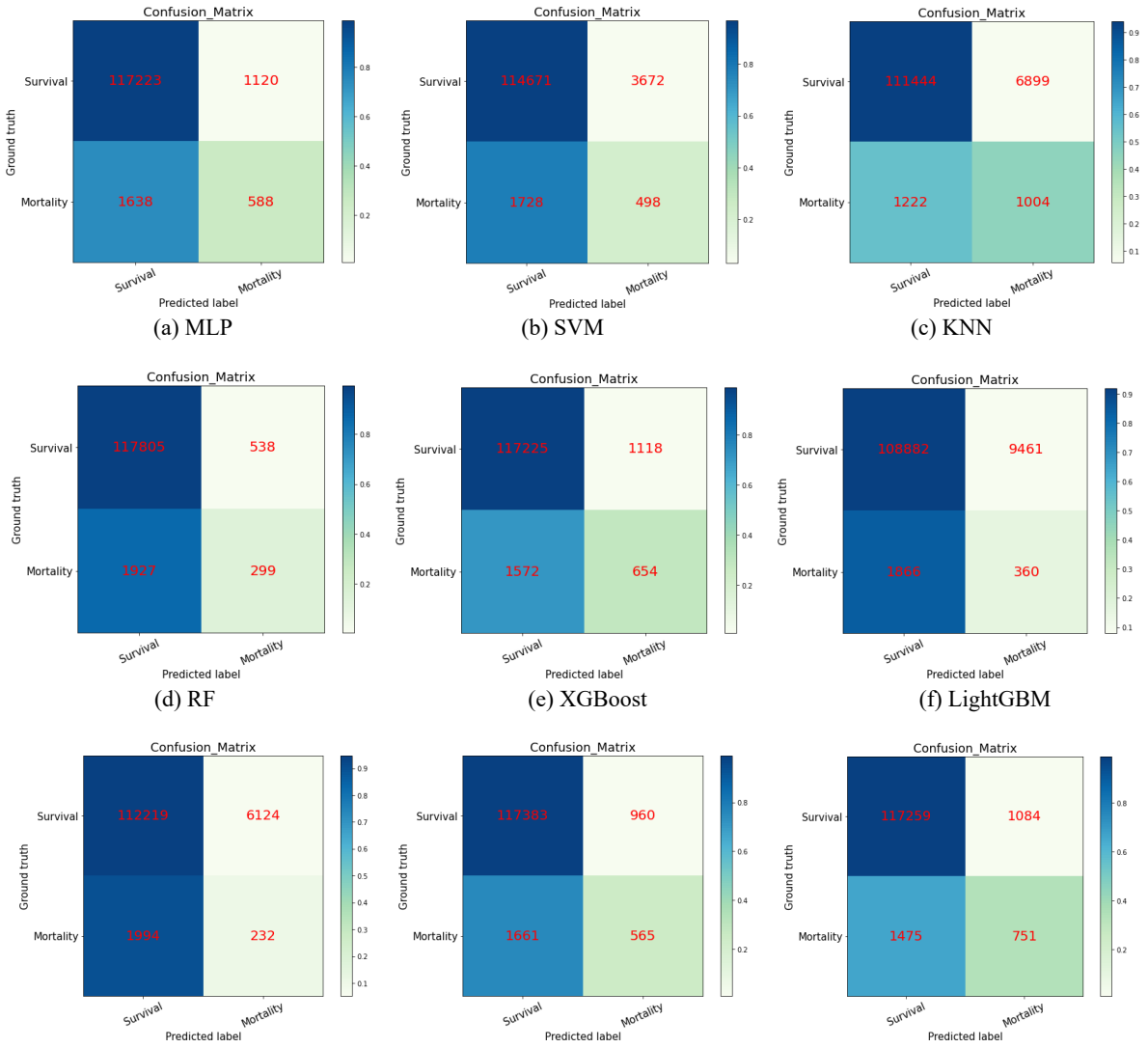


Fig. 4. Training performance of the proposed 1D-MSNet.



(a) MLP (b) SVM (c) KNN

(d) RF (e) XGBoost (f) LightGBM

(g) Transformer (h) 2D-CNN (i) 1D-MSNet

Fig. 5. The test confusion matrices of different methods.

Tables 4 and 5 demonstrate the mortality prediction results of different methods. From Table 4 it can be seen that the proposed approach outperforms the well-known methods on the experimental datasets. After 30 epochs of training, the 1D-MSNet has realized a 98.91% validation accuracy, which is higher than other compared methods except for the Random Forest and 1D-MSNet (halfnorm). However, the Random Forest is an ensemble learning (EL) classifier comprised of multiple decision tree algorithms. Notably, the 1D-MSNet (halfnorm) exhibits comparable performance as the 1D-MSNet. While the distributions differ, the synthetic samples generated by SMOTE maintain the essential patterns and characteristics of the minority class. The large size of the dataset potentially contributes to achieving consistent performance across different distributions, ensuring a balanced augmentation effect. It is also observed that the accuracy of the 1D-MSNet (SE) is close to but slightly worse than that of the proposed method, while the 1D-MSNet (SE) method consumes more time. Especially, as shown in Table 5, the proposed 1D-MSNet outperforms other compared methods and realizes the highest test *ROC-AUC* and *PR-AUC* of 64.19% and 32.41%, respectively. On the test set, the proposed 1D-MSNet architecture achieves top performance relative to other state-of-the-art methods, which

reveals the outstanding capability of the proposed approach for the mortality prediction task. It can also be observed from Figs. 4 and 5. The 1D-MSNet has attained an impressive training performance, and the tested confusion matrix results of the proposed approach are better than that of other compared methods. Besides, it is essential to emphasize that the training time consumption of the 1D-MSNet is 24 minutes, which is less than the average time consumed by the comparison methods of 53 minutes. In contrast, other well-known methods, such as SVM and KNN, take over 3 hours for model training. Therefore, based on the experimental findings, it can be assumed that the 1D-MSNet has presented competitive advantages in both the accuracy and efficiency for the mortality prediction task. The crucial explanation for the substantial effects of the proposed method is that the SMOTE-based data augmentation alleviates the imbalance sample problem. In place of the regular convolution, the multi-scale convolution kernel used in the bottom convolutional layers helps the model extract high-level features and enhance the richness of the convolutional features. Moreover, the enhanced 1D SPP module, featuring atrous and causal convolutions, expands the convolutional receptive field while minimizing computational burden. Simultaneously, this refined 1D SPP module mitigates information loss compared to conventional pooling operations, leading to enhanced model accuracy. Beyond that, the MFL function applied in the network further mitigates the influence of unbalanced samples and enhances the model performance. By contrast, the other methods are frequently-used ML algorithms or single neural networks. Though various optimization and parameter tuning operations are conducted, these methods do not attain desired results. As a consequence, the proposed method achieves the best performance in the comparative experiment of mortality prediction.

Subsequently, we carried out the LoS prediction experiment on the MIMIC-IV v1.0 dataset, and a total of 511,741 rows of records were extracted for the LoS prediction task. Similarly, in addition to preserving 30% of samples from the training dataset as the validation dataset, the training and test sets are split with a ratio of 3:1. To measure the prediction performance of the LoS prediction models, the root mean square error (*RMSE*), mean absolute error (*MAE*), coefficient of determination (R^2 or *R-Square*), and explained variance (E_{VAR}) are chosen as performance metrics, which can be expressed as

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \tag{11}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \tag{12}$$

$$R^2 = 1 - \sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \Big/ \sum_{i=1}^{N}(y_i - \bar{y})^2 \tag{13}$$

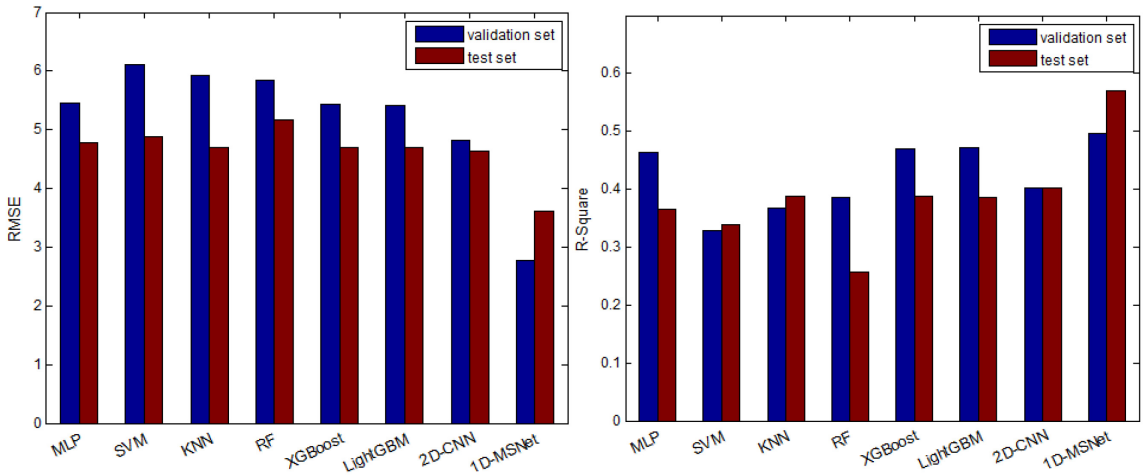
$$E_{VAR} = 1 - var(y_i - \hat{y}_i)/var(y_i) \tag{14}$$

where y_i and $\hat{y}_i$ represent the actual and predicted values, respectively. $var(\cdot)$ signifies the variance function. It is essential to emphasize that for both the E_{VAR} and R^2, the ideal value is 1, while the bigger

value is worse for the *MAE* and *RMSE* metrics.

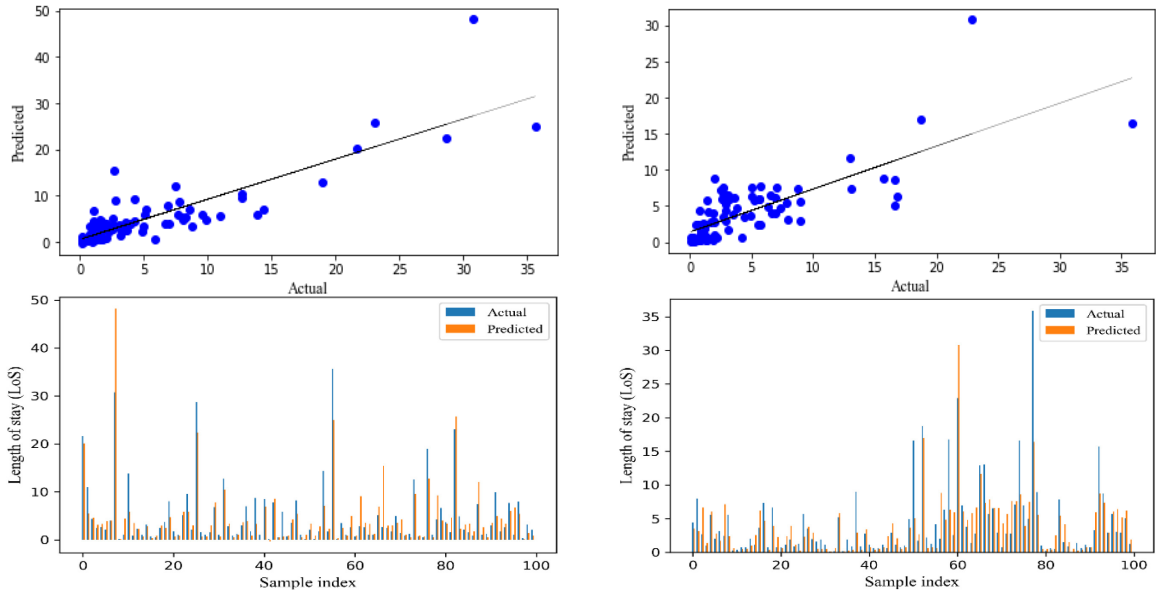
Table 6 LoS prediction accuracy of different methods.

Models	LoS prediction accuracy on the validation set				LoS prediction accuracy on the test set				Time (s)
	$MAE^{\downarrow}$	$RMSE^{\downarrow}$	$R^{2\uparrow}$	$E_{VAR}^{\uparrow}$	$MAE^{\downarrow}$	$RMSE^{\downarrow}$	$R^{2\uparrow}$	$E_{VAR}^{\uparrow}$	
MLP	2.35±0.11	5.46±1.05	0.46±0.07	0.46±0.07	2.83±0.69	4.77±0.91	0.36±0.14	0.41±0.18	0:12:36
SVM	2.25±0.10	6.11±1.02	0.33±0.05	0.34±0.04	2.61±0.86	4.87±1.26	0.34±0.03	0.43±0.11	5:14:53
KNN	2.52±0.11	5.94±1.05	0.37±0.07	0.37±0.06	2.64±0.58	4.69±0.73	0.39±0.18	0.44±0.26	0:13:09
RF	2.53±0.11	5.85±1.07	0.39±0.07	0.39±0.07	2.87±0.75	5.17±1.09	0.26±0.22	0.28±0.23	0:05:36
XGBoost	2.35±0.09	5.43±1.08	0.47±0.07	0.47±0.07	2.75±0.63	4.69±0.84	0.39±0.16	0.41±0.17	0:02:30
LightGBM	2.36±0.08	5.42±1.07	0.47±0.07	0.47±0.07	2.69±0.68	4.69±1.06	0.39±0.06	0.41±0.07	0:02:32
TCNN	2.58±0.02	6.74±0.01	0.35±0.01	0.35±0.01	2.10±0.06	3.65±0.02	0.29±0.01	0.29±0.01	0:05:34
2D-CNN	2.81±0.44	4.82±1.72	0.40±0.01	0.44±0.05	2.62±0.33	4.63±0.37	0.40±0.08	0.44±0.14	0:17:11
1D-MSNet	2.17±0.27	2.78±0.27	0.50±0.11	0.52±0.13	2.42±0.10	3.61±0.14	0.57±0.12	0.59±0.21	0:11:23



(a) *RMSE* (b) *R-Square*

Fig. 6. The *RMSE* and *R-Square* of the 1D-MSNet.



(a) LoS prediction on validation set (b) LoS prediction on test set

Fig. 7. The tested accuracy of the proposed approach.

Table 6 presents the LoS prediction accuracy of different methods on the validation and test sets, and the tested *RMSE* and *R-Square* of the proposed approach are portrayed in Fig. 6. Meanwhile, Fig. 7 depicts the partial prediction samples of LoS on the validation and test sets using scatter and bar plots. From Table 6 it can be visualized that the 1D-MSNet has realized the *MAE* of 2.17 and 2.42, and the *RMSE* of 2.78 and 3.61 on the validation and test sets, which are the optimum values of all the algorithms. Besides, for the *R-Square* indicator, the proposed approach has attained the greatest values of 0.50 and 0.57 on the validation and test sets, which is superior to other compared methods. The proposed approach has also presented a competitive advantage in computation time, which is lower than the average time consumption of the comparison methods. In addition, except for some singularities, the bar charts of the predicted value are basically consistent with that of the actual values, as seen in Fig. 7. The scatter and bar charts of the prediction results also depict the effectiveness and feasibility of the proposed method. Depending upon the experimental analysis, it can be assumed that the 1D-MSNet has outperformed other state-of-the-art methods and delivers impressive efficiency for the LoS prediction task. Furthermore, we have also conducted a performance survey of the proposed method compared with the results reported in the literature, as shown in Table 7. Table 7 reveals that the proposed approach has delivered a comparable result on the MIMIC-IV datasets, which is superior to that of most existing methods. Consequently, the comparative analysis results indicate the competitive advantages of the proposed approach for LoS prediction.

Table 7 Comparison results with recent literature [46].

ID	References	Year	Description	$R^{2\uparrow}$	$RMSE^{\downarrow}$
1	Zimmerman et al. [48]	2006	Mean	0.00	7.23
2	Zimmerman et al. [48]	2006	Median	-0.14	7.71
3	Harutyunyan et al. [49]	2019	LSTM	0.28	6.61
4	Harutyunyan et al. [49]	2019	Multi-Channel LSTM (MC-LSTM)	0.26	6.20
5	Vaswani et al. [50]	2017	Transformer	0.27	6.18
6	Rocheteau et al. [18]	2021	Temporal Pointwise Convolution (TPC)	0.54	4.90
7	Al-Dailami et al. [46]	2022	Temporal Dilated Separable Convolution with Context-Aware Feature Fusion (TDSC-CAFF)	0.64	4.30
8	This study	2023	1D-MSNet	0.57	3.61

3.3 Experiments on other benchmark datasets

To verify the generalization ability of the proposed scheme, we further implemented the mortality risk and LoS prediction experiments on additional two benchmark datasets, including the COVID-19 [51] and Kaggle benchmark datasets (https://www.kaggle.com/datasets/aayushchou/hospital-length-of-stay-dataset-microsoft), respectively. The COVID-19 dataset contains the demographics and clinic diagnosis information of 4,711 patients with confirmed COVID-19 infections. Each patient has a total of 85 features in the dataset, such as age, severity, myocardial infraction (MI), and diabetes mellitus simple (DM simple), etc. It is worth noting that different types of populations are included in the dataset, so this study is not limited to a similar type of patients, and it is generalized towards a larger group of population. Tables 8 and 9 summarize the mortality prediction results of different methods on the COVID-19 dataset. From Table 9 it can be seen that the proposed 1D-MSNet has achieved the highest

test *F1-Score* of 80.24% on the COVID-19 dataset except that of the Random Forest and XGboost ensemble learning algorithms. Particularly, the proposed method realizes the test *ROC-AUC* of 74.99%, which is the best value of all the compared methods.

Table 8 Mortality prediction of different methods on the COVID-19 dataset.

Methods	Training set (%)			Validation set (%)			Time (s)
	Acc↑	*Rec*↑	*F1*↑	*Acc*↑	*Rec*↑	*F1*↑	
MLP	74.51±2.31	74.51±2.31	75.67±1.98	73.09±1.46	73.09±1.46	74.78±1.36	0:00:02
SVM	75.56±0.01	75.56±0.01	65.96±0.01	78.06±0.13	78.06±0.13	69.31±0.07	0:00:03
KNN	83.78±3.49	83.78±3.49	80.73±4.47	77.09±0.25	77.09±0.25	72.09±0.03	0:00:02
RF	89.72±0.77	89.72±0.77	89.14±0.84	83.15±0.93	83.15±0.93	81.38±1.41	0:00:02
XGBoost	89.84±1.46	89.84±1.46	89.14±1.53	84.00±0.36	84.00±0.36	82.34±0.71	0:00:03
LightGBM	74.75±3.93	74.75±3.93	72.58±4.69	78.30±2.79	78.30±2.79	75.25±3.54	0:00:02
TCNN	81.55±0.57	81.55±0.57	80.79±0.22	82.90±1.70	82.90±1.70	82.11±1.21	0:00:41
Transformer	76.09±0.69	76.09±0.69	68.79±3.35	79.39±0.61	79.39±0.61	73.03±0.34	0:03:24
2D-CNN	77.54±1.41	77.54±1.41	71.17±2.95	79.27±0.73	79.27±0.73	73.26±1.58	0:01:17
1D-MSNet	81.67±1.26	81.67±1.26	80.83±1.09	81.81±0.24	81.81±0.24	80.94±0.30	0:01:37

Table 9 The mortality prediction on the COVID-19 test dataset.

Methods	Test set (%)				
	Acc↑	*Rec*↑	*F1*↑	*ROC-AUC*↑	*PR-AUC*↑
MLP	73.83±1.27	73.83±1.27	75.02±1.10	70.62±0.85	35.79±0.88
SVM	76.09±0.07	76.09±0.07	66.37±0.17	51.11±0.14	33.12±0.66
KNN	76.23±0.71	76.23±0.71	71.31±0.99	56.31±1.16	21.40±2.87
RF	83.45±0.63	83.45±0.63	82.03±0.76	71.28±1.11	43.55±1.50
XGBoost	83.73±0.01	83.73±0.01	82.50±0.16	72.26±0.59	44.20±0.23
LightGBM	79.60±1.73	79.60±1.73	74.51±3.61	59.53±3.95	37.63±1.89
TCNN	81.82±2.79	81.82±2.79	79.34±1.30	66.58±2.54	39.96±1.04
Transformer	77.08±1.98	77.08±1.98	70.84±4.01	55.49±4.85	25.23±7.20
2D-CNN	77.86±1.06	77.86±1.06	71.46±2.45	56.11±2.57	30.85±4.08
1D-MSNet	80.97±1.63	80.97±1.63	80.24±1.43	74.99±2.89	41.28±0.65

The LoS prediction experiment is further implemented on the Kaggle benchmark dataset, where 100k data points on patients admitted into hospital, indicators of their health condition, and how long they were admitted in the hospital are included. This dataset was open sourced by Microsoft. It is useful for predicting LoS as a regression task and 27 features are contained in this dataset. Table 10 displays the LoS prediction results of different methods on the Kaggle benchmark dataset. As seen in Table 10, the proposed 1D-MSNet has achieved the highest E_{VAR} of 0.85 on the validation and test sets, respectively. It also achieves favorable results for the remaining metrics. To conclude, the test results reveal that the proposed 1D-MSNet has showcased remarkable performance across additional benchmark datasets, which demonstrates the validity and feasibility of the proposed approach compared with other state-of-the-art methods.

Table 10 LoS prediction on the Kaggle benchmark dataset.

Models	LoS prediction accuracy on the validation set				LoS prediction accuracy on the test set				Time (s)
	$MAE^{\downarrow}$	$RMSE^{\downarrow}$	$R^{2\uparrow}$	$E_{VAR}{}^{\uparrow}$	$MAE^{\downarrow}$	$RMSE^{\downarrow}$	$R^{2\uparrow}$	$E_{VAR}{}^{\uparrow}$	
MLP	0.71±0.11	0.96±0.06	0.83±0.02	0.83±0.01	0.72±0.10	0.97±0.04	0.83±0.02	0.83±0.01	0:00:14
SVM	0.87±0.01	1.18±0.01	0.75±0.01	0.76±0.01	0.87±0.01	1.18±0.01	0.75±0.01	0.76±0.01	0:09:39
KNN	1.02±0.07	1.43±0.11	0.64±0.04	0.66±0.05	1.02±0.06	1.43±0.11	0.63±0.05	0.66±0.05	0:00:18
RF	0.68±0.01	0.96±0.01	0.84±0.01	0.84±0.01	0.69±0.01	0.97±0.01	0.83±0.01	0.83±0.01	0:00:01
XGBoost	0.69±0.20	0.92±0.33	0.84±0.12	0.85±0.06	0.69±0.20	0.92±0.32	0.84±0.12	0.85±0.07	0:00:01
LightGBM	0.75±0.25	0.99±0.30	0.82±0.12	0.82±0.12	1.00±0.24	0.75±0.28	0.82±0.13	0.82±0.13	0:00:01
TCNN	0.69±0.04	0.94±0.03	0.85±0.02	0.85±0.01	0.69±0.04	0.92±0.03	0.84±0.01	0.84±0.01	0:01:27
Transformer	0.76±0.01	1.01±0.05	0.80±0.01	0.81±0.02	0.80±0.01	1.08±0.01	0.78±0.01	0.78±0.01	0:06:04
2D-CNN	0.83±0.07	1.02±0.20	0.81±0.03	0.83±0.02	0.83±0.07	1.02±0.08	0.81±0.03	0.83±0.02	0:01:49
1D-MSNet	0.67±0.09	0.93±0.04	0.85±0.02	0.85±0.01	0.67±0.09	0.93±0.03	0.84±0.02	0.85±0.02	0:02:55

3.4 Ablation study

To analyze the efficacy of the proposed method, we implement the ablation study on the model, where we analyze the behavior of multi-scale filters, SPP module, and atrous causal convolutions on the MIMIC-IV dataset for the LoS prediction task. In the first ablation experiment, we remove the multi-scale filters and uniformly use the filter size of 3 to replace the original filter sizes of 1, 3, and 5 for investigating the model performance. We observe a minor decrease in the result of the ablated model, where the *R-Square* value drops to 0.47 (decreased by 0.03) and 0.44 (decreased by 0.13) on the validation set and test set, respectively. Although the ablation model still outperforms the baseline models, it suffers a notable decline relative to the proposed 1D-MSNet architecture. Subsequently, we conduct the second ablation experiment and remove the ACSPP module from the network. A significant drop in accuracy occurs on the ablation model and its *R-Square* drops to 0.33 (decreased by 0.16) and 0.17 (decreased by 0.40) on the validation and test sets. Also, the *RMSE* value rises to 6.08 (increased by 3.30) and 5.47 (increased by 1.86), respectively. This ablation experiment demonstrates that removing the ACSPP module leads to a notable negative influence on the model accuracy. Further, we evaluate the effect of the atrous and causal convolutions in the ACSPP module. To do so, we remove the atrous and causal convolutions and only use the traditional SPP module to incorporate into the network for comparing models. We notice a decrease in the efficiency where the validation *R-Square* drops to 0.44 (decreased by 0.06) and the test *R-Square* drops to 0.39 (decreased by 0.18). This ablation experiment reveals that removing the atrous and causal convolutions has a significant influence on the performance in comparison with the aggregated ACSPP module of the proposed approach. Table 11 displays the ablation experiment results.

Table 11 Comparison results of ablation experiments.

Ablation approach	LoS prediction accuracy on validation set				LoS prediction accuracy on test set			
	$MAE^{\downarrow}$	$RMSE^{\downarrow}$	$R^{2\uparrow}$	$E_{VAR}{}^{\uparrow}$	$MAE^{\downarrow}$	$RMSE^{\downarrow}$	$R^{2\uparrow}$	$E_{VAR}{}^{\uparrow}$
Delete multi-scale filters	2.39±0.28	5.43±1.42	0.47±0.14	0.47±0.13	2.70±0.47	4.47±0.73	0.44±0.17	0.45±0.17
Delete ACSPP module	2.64±0.21	6.08±1.01	0.33±0.05	0.34±0.05	3.23±0.89	5.47±1.36	0.17±0.05	0.17±0.03
Delete atrous causal	2.55±0.13	5.58±0.96	0.44±0.05	0.45±0.06	2.94±0.80	4.67±0.88	0.39±0.14	0.39±0.13
This study	2.17±0.27	2.78±0.27	0.50±0.11	0.52±0.13	2.42±0.10	3.61±0.14	0.57±0.12	0.59±0.21

4. Conclusions

Estimating the inpatient's LoS and mortality is a challenging daily task for doctors in the healthcare field. To address the challenge, this study proposes a novel one-dimensional CNN architecture, namely 1D-MSNet, to predict patients' status: in-hospital LoS and mortality in the ICU, and generate a state-of-the-art performance for both tasks. In the bottom convolutional layers of the proposed network architecture, the 1D multi-scale convolution kernel is substituted for the single convolution kernel to enlarge the convolutional receptive fields and enhance the richness of the convolutional features. Following the bottom convolutional layers, the atrous causal SPP, which includes 4 parallel 1D average pooling layers concatenated by 4 parallel 1D atrous causal convolutional layers and up-sampling layers, is incorporated into the network for extracting deep-level features. It is noteworthy that the atrous convolution is used to expand the convolutional receptive fields and address long-distance dependencies. Causal convolution is dedicated to handling sequence data and avoiding series information leakage from the future to the past. Besides that, the MFL function is in place of the traditional CE loss function for alleviating the imbalanced-class problem in mortality prediction. The proposed approach is evaluated using publicly accessible databases for LoS and mortality prediction tasks. Experimental results reveal that the proposed 1D-MSNet consistently shows competitive performance on both tasks across a wide range of datasets. We believe that the observed strong and consistent performance is indicative of the underlying robustness of our approach and highlights the potential practical applicability of our method across real-world scenarios. Moreover, by tackling the data imbalance issue, the proposed framework has the potential to mitigate bias.

The integration of ML into healthcare practice and clinical applications holds the potential to yield substantial improvements to the healthcare sector. Nevertheless, building an appropriate DL model is a challenging task due to the dynamic nature, complex model interactions, and variations in real-world problems and data. Moreover, the lack of core understanding turns DL methods into black-box machines that hamper development, which may also raise ethical challenges that need to be seriously considered. In our future work, we will focus on integrating enhanced explainability techniques to make DL models more robust and interpretable, thereby fostering trust between medical professionals and algorithmic recommendations.

Acknowledgments

This research is partially funded by the National Science Foundation under Grant No. 2246158.

Declarations

Conflict of Interests The authors declare no conflicts of interest.

References

[1] Churpek, Matthew M., et al. "Association between intensive care unit transfer delay and hospital mortality: a multicenter investigation." Journal of hospital medicine 11.11 (2016): 757-762.

[2] American Hospital Association. "AHA hospital statistics: fast facts on US hospitals." *American Hospital Association, available at: www/aha/org (accessed May 31, 2017)* (2017).

[3] Grimm, Christi A. "Hospital experiences responding to the COVID-19 pandemic: results of a national pulse survey March 23–27, 2020." *US Department of Health and Human Services Office of Inspector General* 41 (2020): 2020-04.

[4] IHME COVID-19 Health Service Utilization Forecasting Team, and Christopher JL Murray. "Forecasting COVID-19 impact on hospital bed-days, ICU-days, ventilator-days and deaths by US state in the next 4 months." *MedRxiv* (2020): 2020-03.

[5] Tsai, Pei-Fang Jennifer, et al. "Length of hospital stay prediction at the admission stage for cardiology patients using artificial neural network." *Journal of healthcare engineering* 2016 (2016).

[6] Dogu, Elif, Y. Esra Albayrak, and Esin Tuncay. "Length of hospital stay prediction with an integrated approach of statistical-based fuzzy cognitive maps and artificial neural networks." *Medical & Biological Engineering & Computing* 59 (2021): 483-496.

[7] He, Lu, et al. "Neural network-based multi-task learning for inpatient flow classification and length of stay prediction." *Applied Soft Computing* 108 (2021): 107483.

[8] Resar, Roger, et al. "Using real-time demand capacity management to improve hospitalwide patient flow." *The Joint Commission Journal on Quality and Patient Safety* 37.5 (2011): 217-AP3.

[9] Meo, Nicholas, et al. "Introducing an electronic tracking tool into daily multidisciplinary discharge rounds on a medicine service: a quality improvement project to reduce length of stay." *BMJ open quality* 7.3 (2018): e000174.

[10] Cheng, Yu, et al. "Risk prediction with electronic health records: A deep learning approach." *Proceedings of the 2016 SIAM international conference on data mining*. Society for Industrial and Applied Mathematics, 2016.

[11] Wang, Fei, et al. "A framework for mining signatures from event sequences and its applications in healthcare data." *IEEE transactions on pattern analysis and machine intelligence* 35.2 (2012): 272-285.

[12] Peres, Igor Tona, et al. "Data-driven methodology to predict the ICU length of stay: A multicentre study of 99,492 admissions in 109 Brazilian units." *Anaesthesia Critical Care & Pain Medicine* 41.6 (2022): 101142.

[13] Ganaie, Mudasir A., et al. "Ensemble deep learning: A review." Engineering Applications of Artificial Intelligence 115 (2022): 105151.

[14] Mienye, Ibomoiye Domor, and Yanxia Sun. "A survey of ensemble learning: Concepts, algorithms, applications, and prospects." *IEEE Access* 10 (2022): 99129-99149.

[15] Turgeman, Lior, Jerrold H. May, and Roberta Sciulli. "Insights from a machine learning model for predicting the hospital Length of Stay (LOS) at the time of admission." *Expert Systems with Applications* 78 (2017): 376-385.

[16] Alsinglawi, Belal, et al. "Predicting length of stay for cardiovascular hospitalizations in the intensive care unit: Machine learning approach." *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE, 2020.

[17] Fu, Zhiwei, et al. "Predicting the Length of Stay of Patients in Hospitals." *2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, 2021.

[18] Rocheteau, Emma, Pietro Liò, and Stephanie Hyland. "Temporal pointwise convolutional networks for length of stay prediction in the intensive care unit." *Proceedings of the conference on health, inference, and learning*. 2021.

[19] Zhou, Xue, Xin Zhu, and Keijiro Nakamura. "Prediction of Hospitalization Cost and Length of Stay for Patients with Heart Failure Using Deep Learning." *2022 IEEE 4th Global Conference on Life Sciences and Technologies (LifeTech)*. IEEE, 2022.

[20] He, Kaiming, et al. "Spatial pyramid pooling in deep convolutional networks for visual recognition." *IEEE transactions on pattern analysis and machine intelligence* 37.9 (2015): 1904-1916.

[21] Gholamalinezhad, Hossein, and Hossein Khosravi. "Pooling methods in deep neural networks, a review." *arXiv preprint arXiv:2009.07485* (2020).

[22] He, Zewei, et al. "MRFN: Multi-receptive-field network for fast and accurate single image super-resolution." IEEE Transactions on Multimedia 22.4 (2019): 1042-1054.

[23] H. Chen, M. Lin, H. Zhang, G. Yang, G.-S. Xia, X. Zheng, L. Zhang, Multi-level fusion of the multi-receptive fields contextual networks and disparity network for pairwise semantic stereo, in: IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium, IEEE, 2019, pp. 4967–4970.

[24] J. Luo, L. Zhao, L. Zhu, W. Tao, Multi-scale receptive field fusion network for lightweight image super-resolution, Neurocomputing 493 (2022) 314–326.

[25] Ma, Liantao, et al. "Concare: Personalized clinical feature embedding via capturing the healthcare context." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. No. 01. 2020.

[26] Wellner, Ben, et al. "Predicting unplanned transfers to the intensive care unit: a machine learning approach leveraging diverse clinical elements." *JMIR medical informatics* 5.4 (2017): e8680.

[27] Pirracchio, Romain, et al. "Mortality prediction in intensive care units with the Super ICU Learner Algorithm (SICULA): a population-based study." *The Lancet Respiratory Medicine* 3.1 (2015): 42-52.

[28] Lin, Ke, Yonghua Hu, and Guilan Kong. "Predicting in-hospital mortality of patients with acute kidney injury in the ICU using random forest model." *International journal of medical informatics* 125 (2019): 55-61.

[29] González-Robledo, Javier, et al. "Multiclassifier systems for predicting neurological outcome of patients with severe trauma and polytrauma in intensive care units." *Journal of Medical Systems* 41 (2017): 1-8.

[30] Bao, C., F. Deng, and S. Zhao. "Machine-learning models for prediction of sepsis patients mortality." *Medicina Intensiva (English Edition)* (2022).

[31] Ruzicka, Daniel, et al. "Development of a clinical prediction model for recurrence and mortality outcomes after Clostridioides difficile infection using a machine learning approach." *Anaerobe* 77 (2022): 102628.

[32] Ganapathy, Sachit, et al. "Comparison of Bayesian, Frequentist and Machine learning models for predicting the two-year mortality of patients diagnosed with squamous cell carcinoma of the oral cavity." *Clinical Epidemiology and Global Health* 17 (2022): 101145.

[33] Caicedo-Torres, William, and Jairo Gutierrez. "ISeeU2: Visually interpretable mortality prediction inside the ICU using deep learning and free-text medical notes." Expert Systems with Applications 202 (2022): 117190.

[34] Khine, Aye Hninn, Wiphada Wettayaprasit, and Jarunee Duangsuwan. "Ensemble CNN and MLP with nurse notes for intensive care unit mortality." *2019 16th International Joint Conference on Computer Science and Software Engineering (JCSSE)*. IEEE, 2019.

[35] Roopa, Devi EM, et al. "Mortality Prediction of Lung Cancer from CT Images Using Deep Learning Techniques." *2022 Fifth International Conference on Computational Intelligence and Communication Technologies (CCICT)*. IEEE, 2022.

[36] Viton, Fabien, et al. "Multi-channel ConvNet Approach to Predict the Risk of in-Hospital Mortality for ICU Patients." *DeLTA*. 2020.

[37] Theis, Julian, et al. "Improving the in-hospital mortality prediction of diabetes ICU patients using a process mining/deep learning architecture." *IEEE Journal of Biomedical and Health Informatics* 26.1 (2021): 388-399.

[38] Li, Xiaoran, et al. "Deep learning prediction of likelihood of ICU admission and mortality in COVID-19 patients using clinical variables." *PeerJ* 8 (2020): e10337.

[39] Chawla, Nitesh V., et al. "SMOTE: synthetic minority over-sampling technique." *Journal of artificial intelligence research* 16 (2002): 321-357.

[40] Wang, Shutao, et al. "A new method of diesel fuel brands identification: SMOTE oversampling combined with XGBoost ensemble learning." *Fuel* 282 (2020): 118848.

[41] Joshi, Rakesh Chandra, et al. "AI-CardioCare: Artificial Intelligence Based Device for Cardiac Health Monitoring." *IEEE Transactions on Human-Machine Systems* 52.6 (2022): 1292-1302.

[42] Lin, Tsung-Yi, et al. "Focal loss for dense object detection." *Proceedings of the IEEE international conference on computer vision*. 2017.

[43] Johnson, Alistair, et al. "Mimic-iv." *PhysioNet. Available online at: https://physionet. org/content/mimiciv/1.0/(accessed August 23, 2021)* (2020).

[44] Hu, Jie, Li Shen, and Gang Sun. "Squeeze-and-excitation networks." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018.

[45] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." *arXiv preprint arXiv:1412.6980* (2014).

[46] Al-Dailami, Abdulrahman, Hulin Kuang, and Jianxin Wang. "Predicting length of stay in ICU and mortality with temporal dilated separable convolution and context-aware feature fusion." *Computers in Biology and Medicine* 151 (2022): 106278.

[47] Zhang, Ziyuan, et al. "An improved temporal convolutional network with attention mechanism for photovoltaic generation forecasting." Engineering Applications of Artificial Intelligence 123 (2023): 106273.

[48] Zimmerman, Jack E., et al. "Acute Physiology and Chronic Health Evaluation (APACHE) IV: hospital mortality assessment for today's critically ill patients." *Critical care medicine* 34.5 (2006): 1297-1310.

[49] Harutyunyan, Hrayr, et al. "Multitask learning and benchmarking with clinical time series data." *Scientific data* 6.1 (2019): 96.

[50] Vaswani, Ashish, et al. "Attention is all you need." *Advances in neural information processing systems* 30 (2017).

[51] Altschul, David J., et al. "A novel severity score to predict inpatient mortality in COVID-19 patients." *Scientific reports* 10.1 (2020): 16726.

[52] McElroy, Lisa M., and Daniela P. Ladner. "Defining the study cohort: Inclusion and exclusion criteria." *Success in academic surgery: Clinical trials*. London: Springer London, 2013. 131-139.