



*Research article*

## **Predicting the risk of mortality in ICU patients based on dynamic graph attention network of patient similarity**

**Manfu Ma<sup>1</sup>, Penghui Sun<sup>1,\*</sup>, Yong Li<sup>1</sup> and Weilong Huo<sup>2</sup>**

<sup>1</sup> College of Computer Science and Engineering, Northwest Normal University, 967 Anning East Road, Lanzhou 730070, China

<sup>2</sup> College of Traffic and Transportation, Lanzhou Jiaotong University, 88 Anning West Road, Lanzhou 730070, China

\* **Correspondence:** Email: 2021212141@nwnu.edu.cn.

**Abstract:** Predicting the risk of mortality of hospitalized patients in the ICU is essential for timely identification of high-risk patients and formulate and adjustment of treatment strategies when patients are hospitalized. Traditional machine learning methods usually ignore the similarity between patients and make it difficult to uncover the hidden relationships between patients, resulting in poor accuracy of prediction models. In this paper, we propose a new model named PS-DGAT to solve the above problem. First, we construct a patient-weighted similarity network by calculating the similarity of patient clinical data to represent the similarity relationship between patients; second, we fill in the missing features and reconstruct the patient similarity network based on the data of neighboring patients in the network; finally, from the reconstructed patient similarity network after feature completion, we use the dynamic attention mechanism to extract and learn the structural features of the nodes to obtain a vector representation of each patient node in the low-dimensional embedding. The vector representation of each patient node in the low-dimensional embedding space is used to achieve patient mortality risk prediction. The experimental results show that the accuracy is improved by about 1.8% compared with the basic GAT and about 8% compared with the traditional machine learning methods.

**Keywords:** patient similarity network; graph neural network; missing value filling; mortality risk prediction

---

## 1. Introduction

Electronic health records (EHRs) contain a wealth of information on individual patient diagnoses, tests, treatments and outcomes, which can be effectively utilized for clinical prediction studies, such as disease-assisted diagnosis and risk prediction. Predicting the risk of mortality in the ICU is a critical step in the treatment of critically ill patients, and if a patient is at high risk of mortality and stays in the ICU for a long time, a lot of medical resources are used and the burden on the patient's family increases. By predicting the risk of mortality for a specific patient, it can help clinicians identify patients whose condition is likely to deteriorate, so that they can take appropriate treatment measures to prevent this group of patients from worsening and reduce the length of ICU stay.

In recent years, medical research has gradually shifted from a population-based perspective to a personalized perspective, a trend known as precision medicine [1]. Taking inspiration from clinicians, who tend to rely on their past experience in treating similar patients when making clinical judgments, two patients should have similar clinical outcomes if they have similar clinical variables or histological characteristics. Node similarity, based on local information and proposed by some researchers [2], argues that the greater the number of common neighbors that exist between two nodes in a network, the more similar features these two nodes have, i.e., the greater the likelihood that these two nodes belong to the same type of label [3]. However, few people have applied the node similarity problem to patient outcome prediction problems, and according to recent surveys on the application of patient similarity networks in precision medicine or health data processing [4–7], representing data as graphs has the advantage of being highly interpretable and protecting privacy, because patient-specific information cannot be recovered from the similarity metric. Based on these findings, this paper proposes a patient similarity-based dynamic graphical attention network model for predicting patients' mortality risk. Specifically, the similarity relationship between patients is determined based on their clinical data, such as diagnosis, examination and demographics during hospitalization. The two-by-two similarity between patients can be naturally represented as Graph-Patient Similarity Networks (PSNs). Where each node represents a patient and the similarity between patients calculated using clinical data is represented as a weighted edge. Similar patients often have similar treatment trajectories and outcomes, so information from similar patients can often provide powerful support for outcome prediction, disease risk prediction, etc. [6]. This transforms the patient clinical outcome prediction task into a node classification task.

The existence of many cases of unmeasured or unrecorded values of patients in electronic medical record data leads to a high number of missing values in the dataset, and most of the current processing of missing values focuses on methods, such as deletion [8], manual filling, filling of global constants such as N/A and Null and filling of mean or average values [9], which require time and effort on the one hand and low precision on the other. The GAT model [10] takes into account the different importance of different neighboring nodes, assigns different weights to different nodes in the neighborhood, and pays attention to the local structural features of the nodes, but also ignores the non-local information of the nodes, such as the higher-order nodes that are most relevant to that node. Additionally, GAT can only calculate static attention, which has the same attention level for each node, and severely impairs and limits the expressiveness of the attention mechanism [11].

In order to solve the above problems, we propose a Patient Similarity Dynamic Graph Attention Networks (PS-DGAT) based model for mortality risk prediction in ICU, which first constructs a patient similarity network by calculating the similarity of patient clinical data to represent similarities between

patients, then fills in the missing features based on the data of neighboring patients in the network and update the patient similarity network, and finally predicts the risk of mortality of patients by the dynamic GAT model.

The major contributions made in this paper are as follows:

1) We construct a patient similarity network by calculating the similarity of patient clinical data and transforming the structured data into a graph structure, thus representing the relationship between patients more clearly.

2) We use the data of adjacent patients in the network to complete the missing data and reconstruct the patient similarity network.

3) We propose a PS-DGAT model to predict the mortality of ICU patients, and prioritize the vectors with large weights by adjusting the operation order of weighted vectors in GAT to make them better than GAT.

4) We conduct a large number of experiments on the international large-scale public data set MIMIC-III, and compare with the current popular methods to confirm the effectiveness of the proposed model.

## 2. Related work

### 2.1. Mortality risk prediction

Mortality risk prediction is an important task in the medical field and has attracted the attention of an increasing number of researchers in recent years. For example, some researchers [12,13] used traditional machine learning algorithms, such as support vector machines and logistic regression to predict the risk of mortality in heart failure patients, but these methods ignore information about the similarity between patients, resulting in limited prediction performance. To overcome these problems, some researchers have used graph neural networks to predict the risk of mortality in heart failure patients. For example, Lu et al. [14] proposed a graph neural network-based creation of weighted patient networks to extract potential relationships between patients for chronic disease prediction by projecting patient-disease dichotomous graphs. Since traditional GNNs cannot adequately consider the variability of neighboring nodes, the structural information learned is not comprehensive and robust enough [15]. Some other researchers [16] used the GAT model to predict the mortality risk of heart failure patients, achieving adaptive aggregation of different nodes. However, GAT can only calculate static attention, and for each query node, the attention level is the same, which limits its expressive power.

### 2.2. Patient similarity

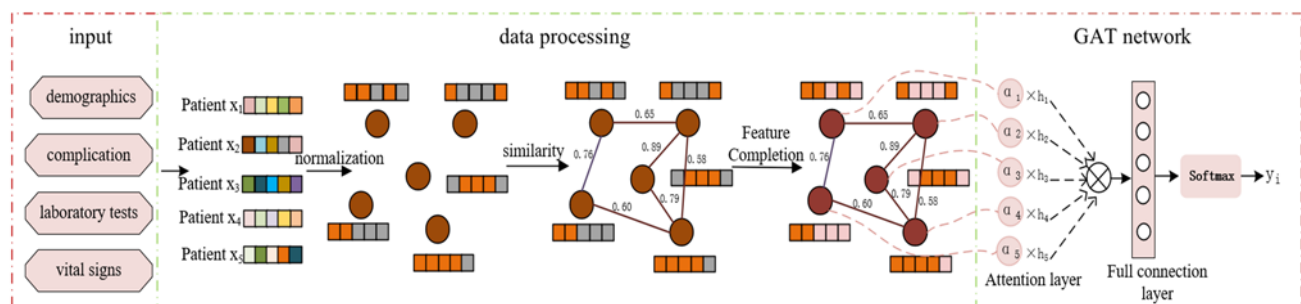
In recent years, an increasing number of researchers have begun to use patient similarity for healthcare data analysis and prediction. Patient similarity analysis simulates the thought process of senior physicians comparing patients, and it refers to selecting clinical concepts (such as diagnosis, symptoms, examination tests, family history, past history, exposure environment, drugs, surgery, genes, etc.) as the characteristic terms of patients in a specific medical setting, quantitatively analyzing the distance between concepts in the semantic space of complex concepts, thus dynamically measuring the distance between patients and screening out the index patient similarity groups [17], so as to assess the current status of patients, predict their prognosis and recommend treatment options. Common

similarity calculation methods include Euclidean distance, cosine similarity, Pearson correlation coefficient, etc. Some researchers have constructed similarity networks by calculating the similarity between patients, for example, in disease prediction, drug treatment, etc. [18,19]. However, these methods calculate patient similarity based on only a single feature or a few features, without considering the combination of patients on multiple features, so it is difficult to reflect the similarity relationship between patients comprehensively and accurately. Moreover, these methods often ignore the missing data, which may lead to inaccurate similarity calculation and thus affect the accuracy of prediction results.

In summary, traditional machine learning methods usually ignore the problem of similarity between patients when predicting their mortality risk. Also, the processing methods for missing values are simpler and cannot fully utilize the clinical data information of similar patients. In addition, despite the rapid development of deep learning in recent years, there is a lack of research in clinical risk prediction using graph neural networks. In this paper, we propose PS-DGAT for predicting the risk of mortality of patients in the ICU. The model uses several features, such as demographic characteristics, laboratory tests and comorbidities to calculate the similarity between patients, and represent the similarity relationship between patients as a patient similarity network. Missing features are filled in by using data from neighbor patients in the network. Finally, patient mortality risk prediction was achieved using a dynamic graph attention network, thus improving the accuracy of the prediction model.

### 3. PS-DGAT model

The detailed architecture of the PS-DGAT model is shown in Figure 1.



**Figure 1.** Model flow chart.

The model consists of the following three modules. 1) Data extraction module, which extracts patient feature data from the MIMIC-III database and screens the data according to the screening criteria. 2) Data processing module, which in the first step preprocesses the extracted data by filling in the missing values and then standardizing them, and then calculates the patient similarity and establishes the connecting edges between patient nodes whose similarity is greater than 0.5, thus constructing a patient similarity network. Then based on patient similarity the filled missing data is processed again by the feature complementation method, which reconstructs the missing features are reconstructed by propagating the known features on the graph. 3) The graph neural network module: The module mainly uses the dynamic attention mechanism to extract features from the patient similarity network reconstructed after feature completion, learns the structural features of the nodes,

and obtains the vector representation of the patient nodes in the low-dimensional embedding space to realize the prediction of patient mortality risk. The following will introduce these three modules and the training process of the PS-DGAT model in detail.

### 3.1. Patient similarity network construction

#### 3.1.1. Patient similarity calculation

**Similarity of binary features.** Gender and comorbidity features are 0, 1 binary features, which can form a binary feature set respectively.  $A$  and  $B$  represent two patients respectively,  $X$  represents the value of a feature corresponding to the patient and the similarity of patients can be calculated according to Eq (1), where the  $S_{bf}(A,B)$  value consistency is 1, and value inconsistency is 0.

$$S_{bf}(A, B) = \begin{cases} 1, & X_A = X_B \\ 0, & X_A \neq X_B \end{cases} \quad (1)$$

**Similarity of numerical features.** Age, BMI, laboratory tests and vital signs are numerical features, which do not involve the time axis after normal normalization, respectively, and we form a 41-dimensional vector to calculate the similarity of numerical features.

$$S_{nf}(A, B) = \sqrt{\sum_i (X_{Ai} - X_{Bi})^2} \quad (2)$$

where  $X_{Ai}$  and  $X_{Bi}$  denote the value of the  $i$ -th feature of the patient,  $i = 1, 2, \dots, 41$ .

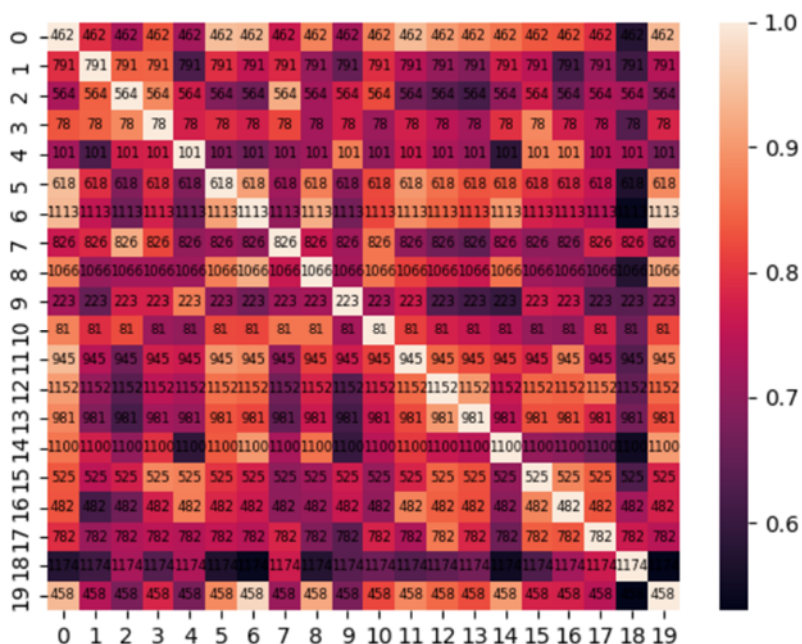
According to the degree of influence of each feature on the patient's outcome, the similarity measure between the two patients was obtained after weighted summation. The calculation formula is as follows:

$$S(A, B) = w_1 \cdot S_{bf} + w_2 \cdot S_{nf} \quad (3)$$

$w_i$  is the weight of feature similarity, satisfying  $w_i \in [0,1]$ ,  $\sum_i w_i = 1$ . Obviously,  $S(A, B) \in [0,1]$ , and the closer to 1, the higher the degree of similarity between patients. Referring to the current relevant research, this paper sets the weights of binary features and numerical features not 0.4 and 0.6 respectively.

#### 3.1.2. Patient similarity network construction

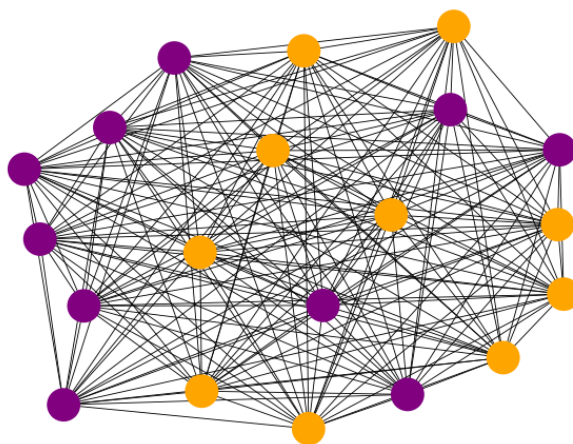
After using the above method to calculate the similarity of the patient nodes, we obtain a similarity matrix, which represents the similarity score between each pair of patient nodes. Due to the large data latitude, we use the sampling method to select the data points with a large number of edges from the high-dimensional similarity matrix as the seed nodes, and use the random walk to sample the probability from the current node according to the weight, select the next node as the sampling result and update the current node as a new node. Repeat this step until a sufficient number of data points are sampled. In order to clearly show the similarity between patients, we sampled 20 data points and plotted the corresponding heat map, label code per cell per patient number.



**Figure 2.** Similarity matrix heat map.

Through the sampled heat map, we can observe the similar patterns and cluster structures between different patients. Although we may lose some details and local information during the sampling process, the simplified heat map still provides meaningful information about the overall similarity relationship. This can help to understand the similarity between patients and provide intuitive visualization.

Through the analysis of hyperparameters, we set the similarity threshold to 0.5, and add an edge between two nodes greater than or equal to 0.5 in the similarity matrix. In this way, we get a weighted patient similarity network. In order to visualize the patient similarity network more intuitively, we use the network diagram to display the 20 important nodes sampled above. As shown in Figure 3, each node represents a patient, and the edge represents the similarity between patients.

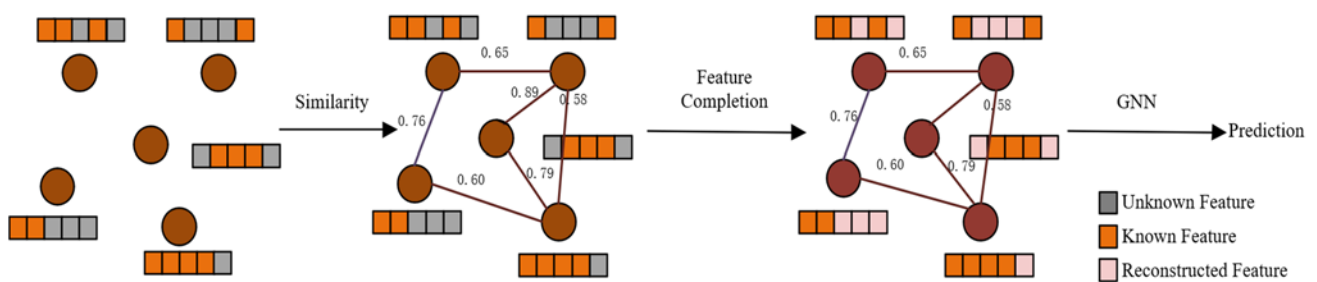


**Figure 3.** Patient similarity network graph.

### 3.2. Missing feature completion

To address the problem of serious missing data, since in node similarity neighboring nodes tend to have similar feature vectors [3], we propose to complete the node features using a patient similarity-based feature completion method. Specifically, the missing features are updated by iteratively propagating the known features of neighboring nodes on the graph to better utilize the relationships between neighboring nodes to update in the missing feature values, resulting in a complete feature matrix. We then recalculated the patient similarity matrix using the patched feature data, which in turn reconstructed the patient similarity network.

Feature completion is aimed at the problem of missing node attributes, while the dataset in this paper has the problem of non-existent edge structure in addition to missing attributes. Therefore, the missing data are first filled using KNN, respectively, to add labels to the missing data, and construct the similarity network using the method in Section 3.1. Then, the feature completion algorithm reconstructs the missing features by iteratively diffusing the known features in the graph. Finally, the patient similarity network is reconstructed with updated node features, which are then fed into the downstream GNN model, and then generates predictions. The propagation process of the algorithm is implemented through the following steps:



**Figure 4.** Feature completion framework.

**Initialization.** Assignment of known feature values (i.e., non-missing values) to the corresponding nodes.

**Propagation.** The missing values are updated by calculating the similarity between neighboring nodes, specifically, the updated values of the node feature vectors are calculated by the following equation:

$$x_i(t+1) = \frac{\text{sum}(S_{ij} * x_j(t))}{\text{sum}(S_{ij})} \quad (4)$$

where  $x_i(t+1)$  denotes the feature vector of node  $i$  at the  $t$ -th iteration;  $S_{ij}$  denotes the similarity between node  $i$  and node  $j$ ;  $x_j(t)$  denotes the feature vector of node  $j$  at the  $t$ -th iteration;  $\text{sum}(S_{ij})$  denotes the similarity sum between neighboring nodes.

**End condition.** Stop feature propagation iteration when the feature vector's no longer changes or changes slightly or reaches the maximum number of iterations (40 times).

**Output result.** Output the node feature matrix  $X'$  after convergence of iterations, where the feature vector of node  $j$  with missing features is reconstructed.

In summary, the flow of the patient similarity-based feature completion algorithm is as follows:

---

**Algorithm 1.** Feature Completion

---

**Input:** node feature vector  $X$ , similarity  $S$   
**Output:** Reconstructed node feature vector  $X'$   
for each vector  $x = 1, 2, \dots, i$  do  
  for neighbor vector  $x' = 1, 2, \dots, j$  do  
    if  $X$  does not converge then  
      Propagation feature:  
       $X = \text{sum}(S \cdot x) / \text{sum}(S)$   
      Reconstruct known features:  
       $X' = X$   
    end if  
  end for  
end for

---

### 3.3. PS-DGAT prediction model

Let the feature matrix of the reconstructed patient similarity network  $\vartheta = (v, \varepsilon)$  after feature completion be  $X$ , denoting the feature vector of node  $i$ . First, the attention coefficients of each patient and its neighbor nodes are calculated, and then the features of the nodes are learned using the dynamic attention mechanism to obtain the vector representation of the patient nodes in the low-dimensional embedding space, and finally the fully connected layer is input for binary prediction of patient mortality or not. The specific operation process is as follows:

**Attention coefficient.** For each patient node  $i$ , the weight vector of  $h_{i,j}$  its neighbor node is calculated, and the attention coefficient of  $a_{i,j}$  each neighbor node is obtained by the softmax function, which is calculated as follows:

$$a_{i,j} = \frac{\exp(\text{Leaky ReLU}(W \cdot [h_{i,j} \parallel h_i]))}{\sum_{k \in N_i} \exp(\text{Leaky ReLU}(W \cdot [h_{i,k} \parallel h_i]))} \quad (5)$$

where  $h_{i,j} \parallel h_i$  denotes the feature vector  $h_i$  of node  $i$  is stitched together with the feature vector  $h_j$  of neighbor node  $j$ .  $W$  is a learnable weight matrix and  $\text{Leaky ReLU}()$  is an activation function.

**Dynamic attention mechanism.** The feature vector  $h'_i$  of node  $i$  is obtained by weighting and summing the feature vectors of neighboring nodes using the attention coefficients  $a_{i,j}$ :

$$h'_i = \sum_{j \in N_i} a_{i,j} \cdot h_j \quad (6)$$

**Feature representation learning.** Using the feature vector  $h'_i$  of node  $i$  as input, mapping it to a vector representation  $z_i$  in a low-dimensional embedding space through a fully connected layer:

$$z_i = \text{ReLU}(W' \cdot h'_i) + b \quad (7)$$



where  $\text{ReLU}()$  is an activation function,  $W'$  is a linear transformation matrix and  $b$  is bias term.

**Prediction.** The low-dimensional embedding vector  $z_i$  of nodes is used as the input of the fully connected layer, and the binary prediction result of whether the patient died is output.

$$y_i = \sigma(W^T h_i + b) \quad (8)$$

where  $y_i$  denotes the prediction result of the node,  $W^T$  denotes the weight,  $b$  denotes the bias and  $\sigma$  denotes the softmax function.

In summary, the process of ICU patient death risk prediction algorithm based on patient similarity dynamic graph attention network is as follows:

---

**Algorithm 2.** Predicting the risk of mortality in ICU patients based on patient similarity dynamic graph attention network

---

**Input:** feature matrix  $X$  of the feature-complemented patient similarity network  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

**Output:** Prediction results for patient nodes  $y_i$

for each node  $i \in \mathcal{V}$  do

Step 1: Calculate the weight vector  $h_{i,j}$  of node  $i$ 's neighbor nodes and get the attention coefficient  $a_{i,j}$ .

Step 2: Use the attention coefficient to weighted sum the feature vectors  $a_{i,j}$  of neighbor nodes to obtain the feature vector  $h'_i$  of node  $i$ .

Step 3: Using the feature vector  $h'_i$  of node  $i$  in the reconstructed patient similarity network as input, it is mapped to the vector representation  $z_i$  in the low-dimensional embedding space through a fully connected layer.

Step 4: The low-dimensional embedding vector  $z_i$  of the node is used as the input of the fully connected layer to output the binary classification result  $y_i$  of whether the patient died.

end for

---

## 4. Experiments

### 4.1. Data extraction and data preprocessing

#### 4.1.1. Data extraction

We used MIMIC-III v1.4, a large publicly available dataset containing approximately 60,000 patients treated in intensive care units (ICUs) from 2001 to 2012 [20], and screened data using the following criteria: 1) patients diagnosed with heart failure according to International Classification of Diseases codes (ICD-9 codes); 2) patients aged  $\geq 18$  years at the time of ICU admission; 3) repeat admissions or repeat ICU admissions of the same patient, and only data from their first ICU admission were included; 4) exclusion of patients with missing N-terminal brain natriuretic peptidogen (NT-proBNP) data. A total of 10,436 patients with a diagnosis of heart failure were queried, and 1255 adult patients were finally included in this study through screening.

Based on previous studies [21–23], we mainly extracted four types of data: 1) demographic characteristics: age, gender, height and weight at admission; 2) comorbidities: including diabetes mellitus, hypertension, chronic kidney disease, chronic pulmonary obstruction, cardiac arrhythmia,

iron-deficiency anemia, hyperlipidemia and atrial fibrillation; 3) vital signs: vital signs within the first 24 hours after admission to the ICU, including temperature, heart rate, blood pressure and respiratory rate; 4) laboratory tests: throughout the ICU stay Laboratory examination index values, including NTproBNP, BNP, troponin, serum creatinine, red blood cells, white blood cells, platelet counts and so on. The coding of patient characteristics is shown in Table 1.

**Table 1.** Information table of patient characteristics.

Index	Feature name	Code
1	ID	Numerical value
2	hadm_id	Numerical value
3	label	0 no dead, 1 dead
4	age	Numerical value
5	sex	1 male, 2 female
6	BMI	Numerical value
7–15	comorbidity	0 no disease, 1 disease
16–22	vital signs	Numerical value
23–51	Laboratory tests	Numerical value

#### 4.1.2. Data preprocessing

After extracting all the features of the patients according to the above criteria, we further processed the data to make it directly usable for the model.

**Missing value processing.** We first use the typical missing value processing method, KNN fill, to initially process the data for missing values. In order to improve the accuracy of the missing value filling, we initially processed the missing values based on the KNN method, we propose the feature completion method to deal with the missing values again by constructing the patient similarity network, and reconstruct the missing features by iteratively diffusing the known features in the graph, so as to further improve the accuracy of the prediction model. The specific method is stated in Section 3.2.

**Data standardization.** We categorized the data into binary and numeric. 1) Binary features included sex and comorbidities, which were assigned values of 1 and 0, respectively, based on whether or not they were male and whether or not they suffered from that comorbidity. 2) Age, BMI, vital signs and laboratory tests were numeric features, and we normalized each feature using the min-max normalization method, which scales the feature values to between 0 and 1, to ensure that they have similar scales. This is shown in Eq (9).

$$X_{\text{normalized}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \quad (9)$$

After preprocessing the data as described above, we use the method in Section 3.1 to calculate patient similarity and construct a patient similarity network. The data size statistics of the final input model are shown in Table 2.

**Table 2.** Data set size statistics.

Dataset	Nodes	Edges	Features	Training	Validation	Test
MIMIC-III	1255	5336	48	1005	125	125

#### 4.2. Experimental setup

In this paper, the experimental data set is divided into training set, verification set and test set according to 8:1:1. A dynamic graph attention network was implemented using the Python programming language and a PyTorch-based deep learning framework to predict the risk of patient mortality. To make the experimental results reproducible, this paper uses SEED = 42 as a random seed to ensure the same randomness for each experiment. The hyperparameters of the training model were `n_epochs = 100`, `batch_size = 128` and `lr = 0.001`. CPU was used for model computation during the training process. To facilitate the observation of the model performance, the classification effect of the model on the test set is measured by three metrics: AUC, Accuracy and F1 score. All experiments were run in the same environment.

#### 4.3. Baseline models

The purpose of this study was to verify the effectiveness of the PS-DGAT model in predicting the risk of death in ICU patients. In order to verify the effect of the model, we compare it with traditional machine learning methods and use different graph convolution layers. The following comparison algorithms are mostly used.

**RandomForest (RF) [24]:** Random forest is an integrated learning method that uses random feature selection and self-sampling techniques to improve model accuracy and generalization by combining multiple decision tree models for classification and regression.

**SVM [25]:** It is a supervised learning algorithm for solving classification and regression problems by constructing optimal hyperplanes to establish decision boundaries between different classes of samples.

**Light GBM [26]:** It is a gradient boosting decision tree algorithm for efficient model training and prediction through learning rate-based decision tree training and histogram algorithm acceleration.

**Decision trees [27]:** are a typical classification method that first processes the data, generates readable rules and decision trees using inductive algorithms and then uses the decisions to analyze the new data. Essentially decision trees are a process of classifying data through a set of rules.

**Multilayer perceptron (MLP) [28]:** is a feedforward artificial neural network model, which consists of input layer, hidden layer and output layer. The working principle is to add a hidden layer or a fully connected hidden layer between the output layer and the input layer, map multiple input data sets to a single output data set and convert the output of the hidden layer through the activation function.

**Graph Convolutional Neural Network (GCN) [29]:** is a multilayer convolutional neural network where each convolutional layer processes only first-order neighborhood information, and by superimposing several convolutional layers, information transfer in multiple-order neighborhoods can be achieved.

**Graph attention network (GAT):** Using attention mechanisms to learn relative importance between nodes and perform node classification or prediction tasks on graph data.

**GraphSAGE [30]:** Learning node representations by aggregating neighboring features of nodes enables efficient node classification and embedding learning on large-scale graph data.

**Cluster-GCN [31]:** Partitioning large-scale graph data into multiple subgraphs using a clustering algorithm and then performing a convolution operation on each subgraph.

#### 4.4. Experimental results

##### 4.4.1. Overall experimental results

In order to verify the effectiveness of our proposed model on the mortality risk prediction task, we compared the overall performance of the model with the baseline models on the MIMIC-III dataset, using Accuracy, AUC and F1 score as evaluation indicators. The comparison results are shown in Table 3.

**Table 3.** Overall performance comparison of ours model and baseline models.

Model	evaluation metrics		
	Accuracy (%)	AUC (%)	F1 score (%)
RF	75.30 ± 0.25	70.53 ± 0.65	55.61 ± 0.29
SVM	80.78 ± 1.22	72.77 ± 0.38	53.77 ± 0.64
Light GBM	76.45 ± 0.30	69.45 ± 0.89	49.90 ± 0.44
Decision Tree	79.20 ± 0.50	53.13 ± 0.21	45.86 ± 0.50
MLP	87.46 ± 0.25	77.98 ± 0.53	60.50 ± 0.32
GCN	87.35 ± 1.15	75.02 ± 0.25	62.28 ± 0.43
GAT	87.96 ± 1.30	70.34 ± 0.67	63.30 ± 0.78
GraphSAGE	87.20 ± 0.88	78.65 ± 0.38	52.86 ± 0.20
Cluster-GCN	86.55 ± 0.34	75.48 ± 0.64	60.88 ± 0.56
<b>PS-DGAT</b>	<b>89.03 ± 0.98</b>	<b>80.92 ± 0.74</b>	<b>69.69 ± 0.23</b>

The above lists the performance of several models on the test set. From the above chart, it can be seen that compared with the other baseline models, the overall effect of the model in the prediction task is better than the traditional machine learning methods and GNN methods, and the Accuracy, AUC and F1 score values are higher than baseline model. The baseline models are categorized into traditional machine learning methods and graph neural network methods, and since the missing feature completion method proposed in this paper is based on graph models, the five machine learning models mentioned above, RF, SVM, Light GBM, Decision Tree and MLP, processed the missing values using only the traditional KNN filling method, and the other graph models use the feature completion method on top of the KNN filling to update the missing values. It can be found that the models that have used the missing feature completion method generally have higher overall performance. Moreover, in order to verify the effectiveness of the missing feature completion method for other graph models, we compare the graph baseline models that used feature completion with those that did not, and the comparison results are shown in Table 4.

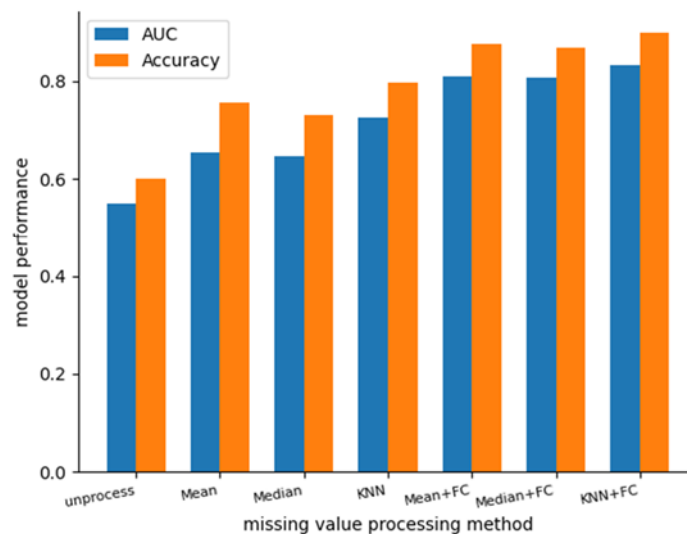
As shown in Table 4, compared to the traditional missing value processing method, our missing feature completion method has improved the performance of all other graph baseline models, and this result shows that the missing feature completion method is effective for the processing of missing features in graph models.

**Table 4.** Comparison of feature completion for baseline models performance improvement.

Model	with feature completion			without feature completion		
	Accuracy (%)	AUC (%)	F1 score (%)	Accuracy (%)	AUC (%)	F1 score (%)
GCN	87.35 ± 1.15	75.02 ± 0.25	62.28 ± 0.43	85.65 ± 0.35	74.53 ± 0.64	60.17 ± 0.13
GAT	87.96 ± 1.30	70.34 ± 0.67	63.30 ± 0.78	86.27 ± 0.23	69.44 ± 0.73	61.58 ± 0.11
GraphSAGE	87.20 ± 0.88	78.65 ± 0.38	52.86 ± 0.20	85.63 ± 1.02	76.88 ± 0.26	50.49 ± 0.70
Cluster-GCN	86.55 ± 0.34	75.48 ± 0.64	60.88 ± 0.56	85.11 ± 0.37	73.28 ± 0.74	58.91 ± 0.22
<b>PS-DGAT</b>	<b>89.03 ± 0.98</b>	<b>80.92 ± 0.74</b>	<b>69.69 ± 0.23</b>	<b>86.20 ± 0.68</b>	<b>79.17 ± 0.52</b>	<b>67.30 ± 0.66</b>

#### 4.4.2. Comparison of missing value handling methods

The processing of missing values is a key step in data preprocessing. The final effect of many models depends largely on the effect of missing values processing. In order to select an effective missing value processing method and to verify the effectiveness of the feature completion method, the prediction results of the model without feature completion and with feature completion methods after the missing values are not processed, mean-completion, median-completion and KNN-completion are compared, and denoted as unprocessed, Mean, Median, KNN, Mean+FC, Median+FC and KNN+FC, respectively. Thus, we verify the necessity and superiority of the feature completion method, and the obtained results are shown in Figure 5.

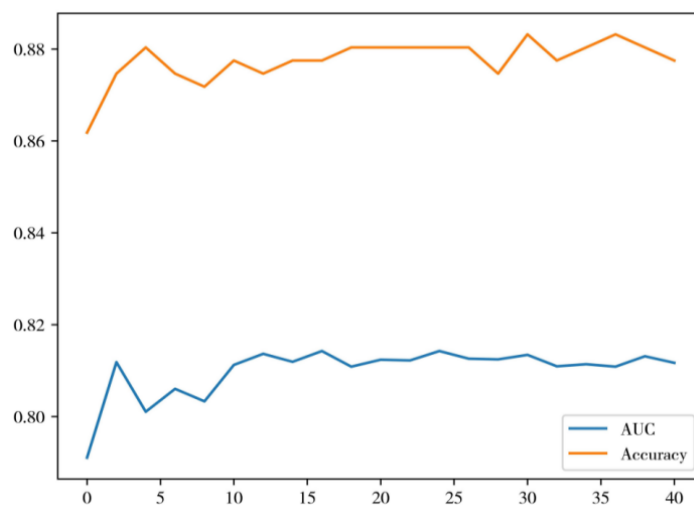
**Figure 5.** Comparison of different missing value processing methods.

Through the experiment results in Figure 5, for the MIMIC-III dataset, the use of KNN fill in this paper is better than other methods, and the feature completion algorithm to fill in the missing values has a significant improvement in the performance of the model and is superior to any traditional missing value processing methods. It shows that the effect of filling the missing features by propagating the information of neighboring nodes is superior to the use of missing value processing methods, such as mean and median. Thus, it verifies the effectiveness of the feature complementation method for the data in this paper.

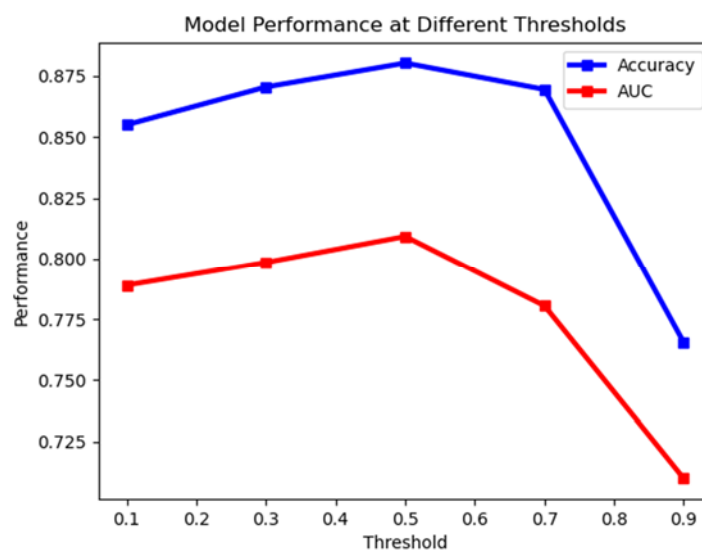
## 5. Discussion

### 5.1. Influence analysis of different iterations of the feature completion method

We reconstruct the missing features by iteratively propagating the known features of adjacent nodes on the graph, and make better use of the relationship between neighbor nodes to fill the missing eigenvalues, so as to obtain a complete feature matrix. Among them, the condition for the feature iteration to stop is that the iteration stops when the feature vector no longer changes or changes slightly or reaches the maximum number of iterations (40 times). We explain the difference in the prediction results of the feature completion method under different iterations, as shown in Figure 6.



**Figure 6.** Effect of iteration times on experimental results.



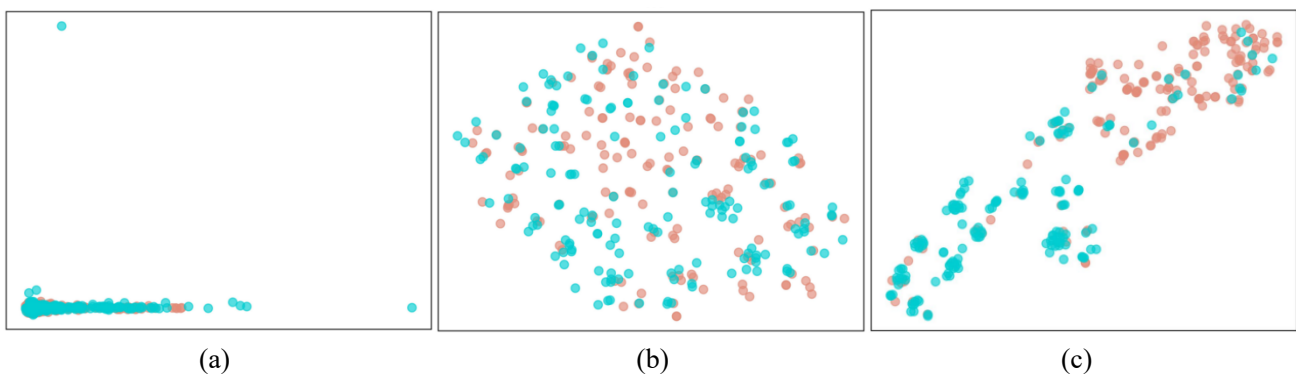
**Figure 7.** Effect of similarity threshold on accuracy.

Figure 6 shows the difference in the prediction results of the feature completion method under different iterations. We observe that with fewer iterations, the prediction results may not make full use of the relationship between neighbor nodes for feature completion, so there may be a large error. As the number of iterations increases, the prediction results gradually converge and stabilize, and the error gradually decreases. However, when a certain number of iterations is reached, further increasing the number of iterations has no obvious effect on the improvement of the results, and the change of the feature vector is very small. Therefore, according to the experimental results and the results of convergence analysis, we chose the maximum number of iterations as 40 times.

### 5.2. Influence analysis of similarity threshold of patient node edges

Because setting different thresholds will produce different network structures, which will affect the classification results of the model, this paper analyzes the effect of setting the similarity threshold between 0 and 1 on the accuracy of the model. The experimental results are shown in Figure 7. From the figure, it can be seen that the threshold value has a significant effect on the experimental results. As the threshold value gradually increases from 0 to 0.5, the classification accuracy of the model also keeps improving. Because the model is classified according to the similarity of nodes, the higher the similarity, the more accurate the classification result. So, when the threshold value increases from 0, more and more less similar nodes are filtered out, and the effective impact of the embedding representation of the node similarity learning nodes based on local information will be smaller and smaller until all similar nodes are filtered out and the experimental results return to the initial state. The horizontal line in the figure indicates the classification accuracy of this paper's model when other parameters are the same. It can be found that the model accuracy is highest when the similarity threshold is set at 0.5, so the threshold is set to 0.5 in this paper.

### 5.3. Dimensionality reduction visualization analysis of PS-DGAT model



**Figure 8.** Visualization results of PS-DGAT model. (a) Initial features; (b) Features reconstructed after feature completion; (c) Features after node embedding.

In order to better understand and explain the effect of the PS-DGAT model, we perform dimensionality reduction visualization analysis on the PS-DGAT model. Using t-SNE [32] dimensionality reduction technology, the patient feature space is mapped to a two-dimensional plane. The relative position and similarity relationship between patients are displayed visually, and the distribution and clustering patterns of patients in the two-dimensional space can be observed.

For Figure 8(a) initial feature distribution, it can be observed that the patient's feature vectors show a certain aggregation trend. This means that in the initial state, the patient's characteristics are relatively consistent in the numerical range, or have similar characteristic patterns. This may be due to the influence of data preprocessing, in which the features are normalized, scaled or outliers are removed. In addition, it is also possible that the patient data has similar pathological state or biological characteristics, resulting in similar feature vectors in the initial state. It should be noted that the aggregation of the initial feature distribution does not necessarily reflect the real differences and similarities between patients. Therefore, we explore the similarity between patients through further dimensionality reduction and visual analysis, and more accurately capture the feature patterns and correlations of patient data.

For Figure 8(b) feature distribution reconstructed after feature completion, it can be observed that the patient nodes show a more dispersed distribution in the dimensionality reduction space, and the distribution of the nodes is relatively uniform. This shows that through the iterative propagation and similarity calculation of neighbor node features, we successfully recover the feature differences between patients and accurately capture the change patterns and correlations of patient data. In the process of feature completion, the feature information of neighbor nodes is effectively transmitted and utilized, so that the patient nodes can be better separated and expanded in the dimensionality reduction space.

For Figure 8(c) feature distribution of nodes after embedding, it can be observed that the distribution of nodes shows the same color aggregation phenomenon, which means that we have successfully completed the classification task of nodes. Nodes with the same color clustering represent that they are classified as the same category or cluster, with similar feature patterns and attributes. This shows that our model effectively captures the similarities and differences between different patients when learning node embedding, so that similar nodes are close to each other in the embedding space, and different nodes are dispersed. Through this node aggregation phenomenon, we can clearly observe the distribution of patient data in the feature space, and further data analysis and interpretation can be performed based on these aggregation results.

## 6. Conclusions

In this paper, we propose a patient similarity-based dynamic graph attention network (PS-DGAT) for predicting the risk of patient mortality in ICU by combining node similarity with graph attention network, and first constructing a patient similarity network by calculating the similarity of patient clinical data to transform the structured data into a graph structure. Then, using similarity, missing features were entered and the patient similarity network was updated based on the data of adjacent patients in the network. Finally, the PS-DGAT model with adjusted order of weighted vector operations in GAT was used to predict the mortality of ICU patients.

In this paper, experimental validation is performed on the MIMIC-III dataset, and the experimental results show that PS-DGAT has better prediction effect than the traditional classification methods, such as GAT, which proves the effectiveness of the model in this paper. The model in this paper also has shortcomings: only structured data are used for clinical data, which may have low prediction accuracy, and unstructured text data should be added in future studies to improve the model performance. In addition, when filling in the missing data in this paper, it is necessary to reconstruct the patient similarity network after reconstructing the features, which increases the complexity, and



better methods can be proposed in subsequent studies to improve the prediction accuracy while reducing the computational complexity and workload.

### Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

### Acknowledgments

We would like to thank the anonymous reviewers for their invaluable comments, whose valuable suggestions and comments contributed significantly to this study.

### Conflict of interest

The authors declare there is no conflict of interest.

### References

1. A. Katznelson, Momentum grows to make ‘personalized’ medicine more ‘precise’, *Nat. Med.*, **19** (2013), 249. <https://doi.org/10.1038/nm0313-249>
2. T. Wang, L. Y. Yin, X. X. Wang, A community detection method based on local similarity and degree clustering information, *Physica A*, **490** (2018), 1344–1354. <https://doi.org/10.1016/j.physa.2017.08.090>
3. Y. Pan, D. H. Li, J. G. Liu, Z. Y. Liang, Detecting community structure in complex networks via node similarity, *Physica A*, **389** (2010), 2849–2857. <https://doi.org/10.1016/j.physa.2010.03.006>
4. S. Pai, G. Bader, Patient similarity networks for precision medicine, *J. Mol. Biol.*, **430** (2018), 2924–2938. <https://doi.org/10.1016/j.jmb.2018.05.037>
5. A. Sharafoddini, J. Dubin, J. Lee, Patient similarity in prediction models based on health data: a scoping review, *JMIR Med. Inf.*, **5** (2017), e6730. <https://doi.org/10.2196/medinform.6730>
6. S. A. Brown, Patient similarity: Emerging concepts in systems and precision medicine, *Front. Physiol.*, **7** (2016), 561. <https://doi.org/10.3389/fphys.2016.00561>
7. L. Dai, H. Zhu, D. Liu, Patient similarity: methods and applications, preprint, arXiv:2012.01976.
8. S. Dey, Y. Wang, R. J. Byrd, K. Ng, S. R. Steinhubl, C. deFilippi, et al., Characterizing physicians practice phenotype from unstructured electronic health records, in *AMIA Annual Symposium Proceedings*, (2017), 514–523.
9. R. S. Somasundaram, R. Nedunchezian, Evaluation of three simple imputation methods for enhancing preprocessing of data with missing values, *Int. J. Comput. Appl.*, **21** (2011), 14–19. <https://doi.org/10.5120/2619-3544>
10. P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, Y. Bengio, Graph attention networks, in *International Conference on Learning Representations*, (2017), 1–12.
11. M. Rahevar, A. Ganatra, T. Saba, A. Rehman, S. A. Bahaj, Spatial–Temporal dynamic graph attention network for Skeleton-based action recognition, *IEEE Access*, **11** (2023), 21546–21553. <https://doi.org/10.1109/ACCESS.2023.3247820>

12. A. Radhachandran, A. Garikipati, N. S. Zelin, E. Pellegrini, S. Ghandian, J. Calvert, et al., Prediction of short-term mortality in acute heart failure patients using minimal electronic health record data, *Biodata Min.*, **14** (2021), 1–15. <https://doi.org/10.1186/s13040-021-00255-w>
13. D. D. Han, F. S. Xu, L. M. Zhang, R. Yang, S. Zheng, T. Huang, et al., Early prediction of in-hospital mortality in patients with congestive heart failure in intensive care unit: a retrospective observational cohort study, *BMJ Open*, **12** (2022), e059761. <https://doi.org/10.1136/bmjopen-2021-059761>
14. H. Lu, S. Uddin, A weighted patient network-based framework for predicting chronic diseases using graph neural networks, *Sci. Rep.*, **11** (2021), 22607. <https://doi.org/10.1038/s41598-021-01964-2>
15. B. Hu, K. H. Guo, X. K. Wang, J. Zhang, D. Zhou, RRL-GAT: Graph attention network-driven multilabel image robust representation learning, *IEEE Internet Things J.*, **9** (2021), 9167–9178. <https://doi.org/10.1109/JIOT.2021.3089180>
16. J. Liu, X. Q. Shang, L. Y. Song, Y. C. Tan, Research progress of graph neural network in complex graph mining, *J. Software*, **33** (2022), 3582–3618.
17. Z. Jia, R. J. Zong, H. L. Duan, Review of patient similarity analysis based on electronic medical records (in Chinese), *Chin. J. Biomed. Eng.*, **37** (2018), 353–366.
18. X. Y. Zhou, F. Huang, X. H. Zhao, W. J. Xiao, W. Zhang, Predicting drug–disease associations through layer attention graph convolutional network, *Briefings Bioinf.*, **22** (2021), bbaa243. <https://doi.org/10.1093/bib/bbaa243>
19. L. Wang, C. Zhong, gGATLDA: lncRNA-disease association prediction based on graph-level graph attention network, *BMC Bioinf.*, **23** (2022), 1–24. <https://doi.org/10.1186/s12859-021-04548-z>
20. A. Johnson, T. Pollard, L. Shen, L. Lehman, M. L. Feng, M. Ghassemi, et al., MIMIC-III, a freely accessible critical care database, *Sci. Data*, **3** (2016), 160035. <https://doi.org/10.1038/sdata.2016.35>
21. C. C. Chiu, C. M. Wu, T. N. Chien, L. J. Kao, C. Li, H. L. Jiang, Applying an improved stacking ensemble model to predict the mortality of ICU patients with heart failure, *J. Clin. Med.*, **11** (2022), 6460. <https://doi.org/10.3390/jcm11216460>
22. C. Luo, Y. Zhu, Z. Zhu, R. Li, G. Chen, Z. Wang, A machine learning-based risk stratification tool for in-hospital mortality of intensive care unit patients with heart failure, *J. Transl. Med.*, **20** (2022), 136. <https://doi.org/10.1186/s12967-022-03340-8>
23. Y. Wei, H. Zou, M. Wang, Q. Zhang, S. D. Li, H. Y. Liang, Mortality prediction among ICU inpatients based on MIMIC-III database results from the conditional medical generative adversarial network, *Heliyon*, **9** (2023), e13200. <https://doi.org/10.1016/j.heliyon.2023.e13200>
24. L. Breiman, Random forests, *Mach. Learn.*, **45** (2001), 5–32. <https://doi.org/10.1023/A:1010933404324>
25. C. Cortes, V. Vapnik, Support-vector networks, *Mach. Learn.*, **20** (1995), 273–297. <https://doi.org/10.1007/BF00994018>
26. G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, et al., Lightgbm: A highly efficient gradient boosting decision tree, in *Proceedings of Advances in Neural Information Processing Systems (NIP 2017)*, (2017), 3146–3154. <https://doi.org/10.1145/3292500.3330665>
27. S. Zhang, H. H. Tong, J. J. Xu, R. Maciejewski, Graph convolutional networks: a comprehensive review, *Comput. Soc. Netw.*, **6** (2019), 1–23. <https://doi.org/10.1186/s40649-019-0069-y>

28. M. C. Olmedo, M. Paegelow, J. Mas, F. Escobar, *Geomatic Approaches for Modeling Land Change Scenarios, An Introduction*, Springer, (2018), 451–455.
29. B. Charbuty, A. Abdulazeez, Classification based on decision tree algorithm for machine learning, *J. Appl. Sci. Technol. Trends*, **2** (2021), 20–28. <https://doi.org/10.38094/jastt20165>
30. W. Hamilton, R. Ying, J. Leskovec, GraphSAGE: Inductive representation learning on large graphs, in *Advances in Neural Information Processing Systems*, (2017), 1024–1034. <https://doi.org/10.5555/3294771.3294858>
31. W. L. Chiang, X. Liu, S. Si, Y. Li, S. Bengio, C. J. Hsieh, Cluster-GCN: An efficient algorithm for training deep and large graph convolutional networks, in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, (2019), 257–266. <https://doi.org/10.1145/3292500.3330648>
32. L. Maaten, G. Hinton, Visualizing data using t-SNE, *J. Mach. Learn. Res.*, **9** (2008), 2579–2605.



AIMS Press

©2023 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)