



A Contrastive Learning-Based Interpretable Prediction Model for Patients with Heart Failure

Jinxiang Zhang, Tianhan Xu, and Bin Li^(✉)

YangZhou University, Yangzhou, China
lb_kmis@yzu.edu.cn

Abstract. Heart failure is a group of complex clinical syndromes due to any structural or dysfunctional abnormality of the heart that results in impaired filling or ejection capacity of the ventricles. Using historical Electronic Health Records (EHRs) to forecast the risk of critical events in heart failure (HF) patients is an important area of research in the field of personalized medicine. However, it is difficult for some machine learning models to predict the risk of critical events owing to data imbalance and poor feature performance in the EHR data of HF patients. While time series-based deep neural networks have achieved excellent results, they lack interpretability. To solve these problems, this study focuses on proposing a deep neural network prediction model of critical events in heart failure patients based on Contrastive learning and Attention mechanism (CLANet). We evaluate our model on a real-world medical dataset, and the experimental results demonstrate that CLANet improves by 2–10% over the conventional methods.

Keywords: Heart failure · Contrastive learning · Critical event prediction · Electronic Health Records

1 Introduction

Heart failure is a worldwide disease that has developed into a worldwide health-care burden [16]. There are currently approximately 26 million Heart failure patients worldwide. The prevalence of HF is estimated to be 1–2%, and the rate in people over 70 years of age in Europe and the United States is more than 10% [13]. Recent epidemiological evidence shows that the prevalence of Heart failure in China has increased by 44% in the last 15 years. More than 9 million people in China have Heart failure. Therefore, early detection of the risk of critical events in HF patients can reduce the cost of medical care and help doctors formulate a more suitable treatment plan based on the prediction results of critical events in HF patients, helping patients prolong their lives [18].

In recent years, some simple and scalable methods inspired by deep learning have been proposed for automatically representing features, such as One-Hot [17] and Skip-Gram [12]. However, these methods often treat each feature as a

discrete and independent word, which introduces the problem of data sparseness, making it difficult for them to capture the latent semantic content between medical variables.

Most previous studies use only a single type of data (e.g., ICD codes or clinical note), or simply concatenate different types of clinical variables into a whole, ignoring the differences between different types of clinical variables [10]. In fact, each type of medical data represents a different health state of patients, so it is necessary to consider the characteristic information contained in each type of medical data separately. However, existing methods largely ignore this phenomenon. Critical event prediction models are primarily designed to help clinicians make clinical decisions. Without interpretability, clinicians cannot determine whether these predictions can be trusted.

To solve the above problems, we have proposed CLANet. When the EHR data of heart failure patients are input, CLANet can mine the semantic information of the same type of medical variables and different types of medical variables through self-attention. Secondly, Bi-LSTM can also be used to capture the temporal dependencies. At the same time, soft-attention is also used to capture variable-level and visit-level attention scores in the patient’s EHR. Finally, contrastive learning was used to compute the similarity of patient pairs. At the same time, the patient representation is used to make predictions about the risk of patient critical events.

In summary, our contributions are as follows:

1. We propose CLANet, an interpretable deep learning model for predicting patient risk of critical events using EHR data from patients with heart failure. In particular, CLANet incorporates multi-layered attention mechanisms that can capture the semantic information. It can also track fine-grained effects of each medical variable and each visit in patient’s medical records.
2. We introduce contrastive learning by constructing a contrastive loss function. This allows the model to perform well with unbalanced data and effectively improves the predictive performance of the model.
3. We conduct an experimental evaluation on a real EHR dataset, and empirically illustrate that CLANet can achieve state-of-the-art performance.

2 Related Work

In recent years, an increasing number of studies have centred around the use of EHR data to predict patient risk of critical events, including mortality prediction [4], readmission [7], ICU transfer [3], and length of stay prediction [1]. Critical event risk prediction models based on longitudinal EHR data fall into two main categories, namely approaches based on machine learning models and approaches based on deep learning models.

2.1 Machine Learning Predictive Models Using EHR Data

Machine learning [14] methods mainly extract features from EHR datasets manually and then make predictions with machine learning models. For example,

Panahiazar et al. [15] designed a risk prediction model using support vector machines, additional trees, logistic regression, decision trees, and random forests.

Despite the promising results obtained with these methods, the results of these models depend heavily on the quality of the features manually selected by the experimenter. The acquisition of these features requires the introduction of expert knowledge and the use of complex statistics to process the data, which makes it difficult to transfer to other application scenarios. Secondly, the performance of machine learning cannot be guaranteed for datasets with sparse and imbalanced data.

2.2 Deep Learning Predictive Models Using EHR Data

Deep learning is capable to automatically extracting features from patient historical EHR data and is being used by an increasing number of researchers. Two variants of deep learning, CNN and RNN are the most commonly used deep learning models. While CNN [5] can automatically extract features and preserve adjacency relationships between input and neighbouring variables, it treats patient EHR data as chronological records and loses the correlation between parts and wholes.

In comparison, RNN [11] has better temporal modelling capabilities and are therefore more widely used. For example, Le et al. [9] proposed an LSTM-based dual memory neural computer (DMNC) to solve the asynchronous multi-view sequence problem, which allows for view interactions and long-term dependencies to be modelled. The model achieved the best results on the MIMIC-III dataset [8]. Although the RNN performs well in predicting critical events, it lacks interpretability, so the attention mechanism is usually used in EHR-based temporal prediction models. RETAIN [2] is a well-known interpretable prediction model, which consists of two recurrent neural networks and attention mechanism to learn forward and backward representations of patients respectively. Self-attentive and soft-attentive mechanisms are also used in our proposed CLANet.

3 Methodology

The task of predicting critical events in patients with heart failure is mainly divided into two tasks: mortality prediction and ICU transfer prediction. Our proposed model (CLANet) predicts the following four tasks: 48-hour mortality, 7-day mortality, in-hospital mortality and ICU transfer.

3.1 Data Processing

The EHR data used in this paper contain five main types of data. The EHR data of different patients are not the same. Therefore, it is necessary to process the data. If one-hot coding is used to directly represent patients, it will cause data redundancy, because in the MIMIC-III dataset, except for heart failure, there are a total of 3583 diseases in heart failure patients, most of which have

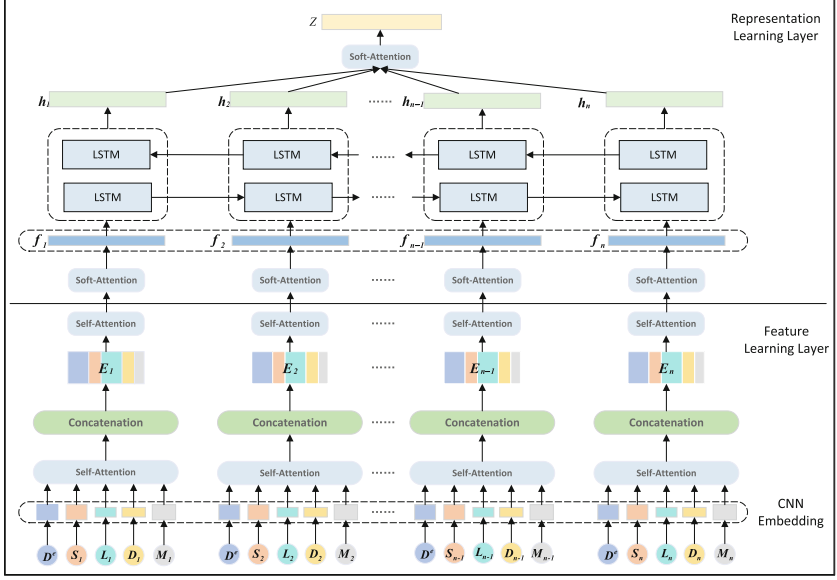


Fig. 1. The architecture of representation learning module.

only been diagnosed in a few patients and are easily treatable. Medication and surgery have the same problem (Fig. 1).

To solve the above problems, the method used in this paper is to first extract the heart failure patients in the dataset and divide them into two categories, namely, death and survival. Death is marked as positive and survival is marked as negative. Then the diagnosis, medication and surgery information in the EHR of the patients were extracted, and these variables of each patient were spliced into sentences, each variable was equivalent to a word, and the same type of patients were spliced into an article. Then the importance of each diagnosis, drug and surgery was calculated from the article using the idea of TF-IDF. The formula is as follows:

$$imp_i = \frac{po_i}{\sum_j po_i} \times \log \frac{|POS| + |NEG|}{sn_i} + \frac{pn_i}{\sum_j pn_i} \times \log \frac{|POS|}{sp_i + 1} \quad (1)$$

where po_i represents the number of occurrences of the variable V_i in POS and NEG, $\sum_j po_i$ represents the sum of all variables appearing in POS and NEG, sn_i represents the sentences containing variable V_i in the articles POS and NEG. pn_i represents the number of occurrences of the variable V_i in POS, $\sum_j pn_i$ represents the number of all variables in POS, and sp_i represents the total number of sentences containing variable V_i in POS. In this paper, the basis of the original TF-IDF is improved. Variables that score higher in positive examples are given more weight.

3.2 Representation Learning Module

The representation learning module mainly includes a feature learning layer and a representation learning layer. The feature learning layer mainly learns the features in the patient's EHR data through 1D-CNN and self-attention. The representation learning layer obtains the patient representation mainly through the soft-attention mechanism and Bi-LSTM.

Feature Learning Layer. The first layer of the feature learning layer uses the information from each patient visit to obtain the embedding of the variables. There are two main reasons for using 1D-CNN to represent each medical variable: firstly, discrete medical variables cannot be directly used for deep learning networks. Secondly, the direct use of one-hot to represent patient visit information does not well capture the correlation between different medical variables.

We first use 1D-CNN to embed each medical variable, specifically, the input vector $V_i = \langle v_i^1, v_i^2, \dots, v_i^m \rangle$, Where m represents the number of different examinations for patients. Then the relevant information embedding R_t of the patient can be obtained by the following formula:

$$R_t = \text{ReLU} \left(\sum_{t=1}^M W_e v_t + b_e \right) \quad (2)$$

where v_t represents the feature of the t th timestamp, W_e is the convolution kernel, and b_e is the bias parameter.

The second layer of the feature learning layer is the self-attention layer, which performs self-attention on each medical data to obtain the internal relationship of each type of medical variable. A representation of each visit in the EHR of HF patients is then obtained. Firstly, the representation vector R_t obtained by 1D-CNN is used as input, and the self-attention mechanism constructs a key matrix K_i , query matrix Q_i and value matrix V_i according to Eq. (3), where KW_i , QW_i and VW_i are trainable weight parameter matrices. Then, the attention weight matrix A_i is computed according to Eq. (4), where d_k represents the latitude of the input vector. Finally, the output matrix O_i is computed as a weighted sum according to Eq. (5).

$$K_i = KW_i R_i, Q_i = QW_i R_i, V_i = VW_i R_i \quad (3)$$

$$A_i = \text{softmax} \left(\frac{Q_i K_i^T}{\sqrt{d_k}} \right) \quad (4)$$

$$O_i = V_i A_i \quad (5)$$

$$S_i^j = \sum_{k=1}^m v_i^k \alpha_i^{jk} \quad (6)$$

where $A_i = \langle a_i^1, a_i^2, \dots, a_i^m \rangle$, $a_i^j = \langle \alpha_i^{j1}, \alpha_i^{j2}, \dots, \alpha_i^{jm} \rangle$, α_i^{jk} is the influence of variable v_i^k on variable v_i^j . $S_i = \langle s_i^1, s_i^2, \dots, s_i^m \rangle$, where s_i^j is the output

matrix of r_i^j . It can be calculated by formula (6). We end up with five output vectors.

Then, the obtained output vectors are concatenated to obtain the joint representation E_i of the patient, and then the contextual semantic relationships of different types of data are obtained through the self-attention mechanism. Finally, the output vector N_i is obtained by the following formula:

$$K_i = KW_i E_i, Q_i = QW_i E_i, V_i = VW_i E_i \quad (7)$$

$$N_i = \text{softmax} \left(\frac{Q_i K_i^T}{\sqrt{d_k}} \right) V_i \quad (8)$$

Representation Learning Layer. It contains three sub-layers: variable-level attention, Bi-LSTM, visit-level attention. variable-level attention uses the output vector $N_i = \langle N_i^1, N_i^2, \dots, N_i^n \rangle$ obtained by the feature learning layer to compute the contribution of medical variable in a visit. Soft-attention is then utilized to obtain the contribution of the medical variables to the visit. The formula is as follows:

$$\alpha_i^j = \text{softmax} \left(g_i^T \tan h \left(CW_i n_i^j \right) \right) \quad (9)$$

$$f_i = \sum_{j=1}^m \alpha_i^j n_i^j \quad (10)$$

where CW_i is the trainable weight parameter matrix and g_i is the trainable context vector. According to Eq. (9), the contribution of each medical variable is calculated. Finally, the contribution of all medical variables is aggregated. Obtain the representation vector f_i of patient visits.

Visit-level attention uses the visit vector obtained earlier and computes the importance of each visit. The temporal dependence between visits is first captured by Bi-LSTM, and then the importance of each visit is computed using soft-attention. This is done primarily because patient visits are a sequential process. Specifically, the patient's visit vector f_i is first fed into a Bi-LSTM in chronological order, which is made up of a forward network and a backward network that can make full use of the information from the past and the future. The forward LSTM reads f_1 to f_t and computed the forward hidden state sequence $\langle \vec{h}_1, \vec{h}_2, \dots, \vec{h}_t \rangle$, given the input vector \vec{f}_i and the previous hidden state \vec{h}_{i-1} . The hidden state is computed by the following formula:

$$\vec{h}_i = \text{LSTM} \left(\vec{f}_i, \vec{h}_{i-1} \right) \quad (11)$$

Similarly, the backward LSTM reads the embedding vector sequence in reverse order, generating the backward hidden state sequence $\langle \overleftarrow{h}_1, \overleftarrow{h}_2, \dots, \overleftarrow{h}_t \rangle$. The hidden state h_i is a combination of forward and backward hidden states, calculate by the following formula:

$$h_i = \left[\vec{h}_i; \overleftarrow{h}_i \right] \quad (12)$$

Then, soft-attention uses a linear transformation normalized by softmax to calculate attention weights according to Eq. (13). Finally, the representation vector z representing the patient is calculated by summarizing the importance of the medical variables for the heart failure patient according to Eq. (14), where w is a trainable vector, b is a trainable scalar, and β represents the importance of the hospital visit h_i .

$$\beta = \text{softmax}(w^T h_i + b) \quad (13)$$

$$z = \sum_{i=1}^n \beta_i h_i \quad (14)$$

3.3 Contrastive Learning Layer

We introduce contrastive learning [6] to improve the classification ability of the model and solve the sample Sparse data problem in the experimental data. First, we construct the sample pair, if their labels are consistent, the label is set to 0, otherwise the label is set to 1. The representation learning module is used to generate the patient representation, and then the similarity of the patient representation in each sample pair is calculated. The following contrastive loss function is used to bring patients with consistent labels closer together and to pull patients with inconsistent labels apart.

$$\mathcal{L}_{\text{Dist}} = (1 - Y) \frac{1}{2} (D_W)^2 + (Y) \frac{1}{2} \{\max(0, m - D_W)\}^2 \quad (15)$$

where Y is the label. D_w is the Euclidean distance between the patient representations of the model's output. The max function takes 0 or the margin m minus the maximum in the distance. In this experiment, the $m = 1$.

3.4 Loss Function

To obtain appropriate model parameters and predict critical events of heart failure patients, a sigmoid function was used to predict the labels of patients, and the cross-entropy between real visit label y_i and the predicted visit label \hat{y}_i was used as the prediction loss function:

$$\mathcal{L}_{\text{CE}} = -\frac{1}{2N} \sum_{i=1}^N y_i^T \log \hat{y}_i + (1 - y_i)^T \log(1 - \hat{y}_i) \quad (16)$$

where y_i is the true label of the i th heart failure patient. \hat{y}_i is the score of the i th patient calculated by CLANet. We use the adma optimizer to optimize the above formulation.

Since the contrastive learning module and the prediction of critical events in heart failure patients can mutually benefit from joint training to obtain the clustered patient representation and the prediction of critical events in heart failure patients, the loss can be expressed as follows:

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda_{\text{Dist}} \mathcal{L}_{\text{Dist}} \quad (17)$$

The contrastive loss function is scaled by a non-negative hyperparameter λ . In this experiment, the hyperparameter $\lambda = 0.5$.

4 Experiments

4.1 Dataset Description

The EHR dataset used in this experiment is the MIMIC-III [8]. Firstly, the data contains demographics, medications, laboratory tests, surgical codes, diagnosis codes. In this study, a variety of information about the patient is used, because whether it is age, laboratory tests, diagnosis, or drug and surgical information, it is essential to predict the health status of the patient. Secondly, because this paper is the critical event prediction of heart failure patients, so 10436 patients diagnosed with heart failure were extracted.

4.2 Implement Details

In this study, machine learning methods are mainly implemented using scikit-learn. All deep learning models in this study were implemented using tensorflow 2.6.0 and all methods used adam optimizer with learning rate set to 0.001. A computer with 90 GB of RAM and a Tesla A40 GPU was used for training. The batch-size was set to 512 for all deep learning models. To avoid overfitting, we introduce dropout strategy and Dropout rate is 0.5. At the same time, the early stopping strategy and L2 regularization are also used. For the proposed CLANet, the embedding size of 1D-CNN for each variable is 256, and the hidden units of LSTM are 30. We randomly split the training, validation and test sets into 0.7 : 0.2 : 0.1. Three measures were used to evaluate the performance of the model: Accuracy, F1-score, and AUC. For all models, we repeat the experiments 20 times and report the average evaluation metric of the test performance.

4.3 Baselines

LR: It is a generalized linear regression analysis model, which is part of the supervised learning in machine learning.

RF: It is a form of ensemble learning, combining many decision trees into a forest.

XGBOOST: It is a decision tree based ensemble algorithm for classification and regression problems.

Bi-LSTM: It is an important variant of deep learning that can handle sequence problems well.

Diople [11]: It is a Bi-RNN model for diagnostic prediction tasks with an attention mechanism that represents a patient's visit as a series of unordered sets composed of multiple unique medical codes.

Retain: It is a combination of two recurrent neural networks and an attention mechanism to learn forward and backward representations of patients respectively, and visit-level weights and variable-level weights can be obtained.

IoHAN [4]: It is an interpretable outcome prediction model based on hierarchical attention, which obtains variable-level and visit-level attention of patients.

Table 1. Performance of Baselines and CLANet on Four key Event Prediction Tasks.

Model	48 h			7day			in-hospital			ICU Transfer		
	Acc	F1	AUC	Acc	F1	AUC	Acc	F1	AUC	Acc	F1	AUC
LR	0.727	0.705	0.756	0.725	0.741	0.747	0.731	0.761	0.750	0.798	0.802	0.876
RF	0.756	0.739	0.666	0.763	0.742	0.669	0.779	0.782	0.669	0.819	0.806	0.809
XGBOOST	0.763	0.750	0.671	0.774	0.766	0.694	0.793	0.763	0.702	0.821	0.803	0.842
Bi-LSTM	0.788	0.796	0.726	0.792	0.808	0.752	0.806	0.822	0.758	0.818	0.805	0.879
Diople	0.804	0.821	0.787	0.812	0.822	0.797	0.817	0.826	0.794	0.813	0.806	0.880
Retain	0.813	0.822	0.786	0.806	0.818	0.805	0.819	0.823	0.795	0.814	0.806	0.885
IoHAN	0.817	0.811	0.791	0.827	0.834	0.810	0.827	0.835	0.796	0.819	0.801	0.886
CLANet	0.832	0.829	0.808	0.826	0.844	0.810	0.843	0.841	0.815	0.841	0.811	0.893

4.4 Result Analysis

Table 1 presents the average performance of the proposed CLANet and other baseline models on the four tasks. It can be seen that CLANet exhibits stable and excellent performance. And we achieve state-of-the-art performance on most metrics.

We first focus on classical machine learning methods, including LR, RF and XGBOOST. Machine learning methods generally show lower performance compared to deep learning methods. The main reason is that they cannot model a patient’s visit as a sequence, but only the patient’s visit sequence as a whole. On the mortality prediction task, the machine learning model achieves 5% lower F1-score and AUC scores than deep learning baselines such as Bi-LSTM. However, on the ICU Transfer task, machine learning methods perform no worse or even better than many deep learning-based methods. The reason for this phenomenon may be that individual signals may be more important than timing information for ICU Transfer tasks. In this case, deep learning models such as Bi-LSTM may suffer from overfitting. Through the attention mechanism and contrastive learning, CLANet can extract the medical variables that have a key impact on the outcome, and can distinguish the representations of patients, which can achieve better prediction results.

As an ordinary deep learning model, the performance of Bi-LSTM is stable, and the performance gap between the Bi-LSTM model and any other deep learning models on the ICU Transfer task is not large. This is mainly because heart failure is a chronic disease, and for the ICU Transfer task, simple deep learning models can capture key variables easily. However, it did not perform as well on the mortality prediction task. For the mortality prediction task, the performance of all attention-based models is outstanding, mainly due to the fact that deep learning can capture key variables in time-series information, and then

amplify these medical variables through the attention mechanism. Although both RETAIN and IoHAN use hierarchical attention, contrast learning is introduced in our model to better distinguish patient representations and achieve the best classification results. In particular, on the ICU Transfer task, CLANet achieves the optimal performance on each metric.

4.5 Ablation Study

In this section, we focus on the comparison between CLANet and its variants that change parts of the full CLANet model. The setup is the same as the previous experiment, but this time we run it 5 times to get the average performance.

CLANet-TF: It is a variant of CLANet without the critical code extraction module and directly using all medical variables.

CLANet-SEA: It is a variant of CLANet without self-attention. The final representation of the patient is directly obtained through variable-level and visit-level attention and Bi-LSTM.

CLANet-SOA: It is a variant of CLANet without soft-attention. Specifically, it directly uses self-attention and Bi-LSTM to obtain the final representation.

CLANet-ATT: It is a variant of CLANet without attention mechanism. Specifically, it directly uses Bi-LSTM to obtain the final representation.

CLANet-CL: It is a variant of CLANet without contrastive learning, specifically, it does not construct sample pairs and directly uses the cross-entropy loss to predict the critical event risk.

Table 2. Average Performance for CLANet’s Variants.

Model	48 h			in-hospital			ICU Transfer		
	Acc	F1	AUC	Acc	F1	AUC	Acc	F1	AUC
CLANet-TF	0.752	0.725	0.729	0.785	0.720	0.796	0.807	0.794	0.825
CLANet-SEA	0.815	0.820	0.804	0.823	0.827	0.789	0.811	0.796	0.889
CLANet-SOA	0.809	0.806	0.795	0.802	0.820	0.785	0.821	0.805	0.882
CLANet-ATT	0.792	0.807	0.759	0.801	0.816	0.796	0.814	0.801	0.877
CLANet-CL	0.819	0.809	0.793	0.820	0.828	0.801	0.809	0.793	0.858
CLANet	0.832	0.819	0.808	0.843	0.841	0.825	0.841	0.811	0.893

The experimental results are presented in Table 2. It can be seen that after the deletion of TF-IDF, the model performance significantly decreases, mainly because the patient visit information is composed of medical variables. The long-tailed distribution of each medical variable may cause redundancy, so this paper uses the keyword extraction method commonly used in NLP to extract key influencing variables and improve the prediction performance of the model. In the mortality prediction task, CLANet-ATT performs the worst and CLANet

performs the best. This shows that all sub-layers of CLANet contribute to the final critical event risk prediction.

Secondly, the performance of CLANet-SEA is better than CLANet-SOA, indicating the effectiveness of variable-level and visit-level attention, mainly because the patient information in MIMIC-III is mainly from icu, and the collected information is not rich enough, and the contextual information is relatively fixed, so the contextual information of patient visit information is not obvious enough. However, CLANet-SOA performs better than no attention mechanism, indicating that the self-attention layer improves the predictive ability of the model. Finally, it can be seen that when the contrastive learning module of the model is removed, the predictive ability of the model is significantly decreased, suggesting that contrastive learning has a facilitating effect on critical event prediction. Especially in ICU Transfer task, due to the simple task, the deep learning model has difficulty learning useful knowledge, which is prone to cause overfitting. Therefore, when we remove contrast learning, the model effect will reach the minimum.

5 Conclusions

Predicting the risk of critical events in HF patients using EHR data is one of the key issues in medical event prediction. The existing critical event prediction models cannot solve the problem of multi-data fusion and interpretability well. To solve the aforementioned problems, we propose CLANet, a multi-layer attention mechanism model based on contrastive learning. Experimental results on MIMIC-III show that CLANet outperforms existing models in terms of prediction performance.

References

1. Cai, X., et al.: Real-time prediction of mortality, readmission, and length of stay using electronic health record data. *J. Am. Med. Inform. Assoc.* **23**(3), 553–561 (2016)
2. Choi, E., Bahadori, M.T., Sun, J., Kulas, J., Schuetz, A., Stewart, W.: Retain: An interpretable predictive model for healthcare using reverse time attention mechanism. In: *Advances in Neural Information Processing Systems*, vol. 29 (2016)
3. Chou, C.A., Cao, Q., Weng, S.J., Tsai, C.H.: Mixed-integer optimization approach to learning association rules for unplanned ICU transfer. *Artif. Intell. Med.* **103**, 101806 (2020)
4. Du, J., et al.: An interpretable outcome prediction model based on electronic health records and hierarchical attention. *Int. J. Intell. Syst.* **37**(6), 3460–3479 (2022)
5. Feng, Y., et al.: Patient outcome prediction via convolutional neural networks based on multi-granularity medical concept embedding. In: *2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 770–777. IEEE (2017)
6. Hadsell, R., Chopra, S., LeCun, Y.: Dimensionality reduction by learning an invariant mapping. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006)*, vol. 2, pp. 1735–1742. IEEE (2006)

7. He, D., Mathews, S.C., Kalloo, A.N., Hutfless, S.: Mining high-dimensional administrative claims data to predict early hospital readmissions. *J. Am. Med. Inform. Assoc.* **21**(2), 272–279 (2014)
8. Johnson, A.E., et al.: MIMIC-III, a freely accessible critical care database. *Sci. Data* **3**(1), 1–9 (2016)
9. Le, H., Tran, T., Venkatesh, S.: Dual memory neural computer for asynchronous two-view sequential learning. In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1637–1645 (2018)
10. Luo, J., Ye, M., Xiao, C., Ma, F.: HiTANet: hierarchical time-aware attention networks for risk prediction on electronic health records. In: *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 647–656 (2020)
11. Ma, F., Chitta, R., Zhou, J., You, Q., Sun, T., Gao, J.: Dipole: diagnosis prediction in healthcare via attention-based bidirectional recurrent neural networks. In: *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1903–1911 (2017)
12. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. *arXiv preprint [arXiv:1301.3781](https://arxiv.org/abs/1301.3781)* (2013)
13. Mosterd, A., Hoes, A.W.: Clinical epidemiology of heart failure. *Heart* **93**(9), 1137–1146 (2007)
14. Nistal-Nuño, B.: Developing machine learning models for prediction of mortality in the medical intensive care unit. *Comput. Methods Programs Biomed.* **216**, 106663 (2022)
15. Panahiazar, M., Taslimitehrani, V., Pereira, N., Pathak, J.: Using EHRS and machine learning for heart failure survival analysis. *Stud. Health Technol. Inform.* **216**, 40 (2015)
16. Ponikowski, P., et al.: Heart failure: preventing disease and death worldwide. *ESC Heart Failure* **1**(1), 4–25 (2014)
17. Uriarte-Arcia, A.V., López-Yáñez, I., Yáñez-Márquez, C.: One-hot vector hybrid associative classifier for medical data classification. *PLoS One* **9**(4), e95715 (2014)
18. Wang, Y., et al.: Early detection of heart failure with varying prediction windows by structured and unstructured data in electronic health records. In: *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 2530–2533. IEEE (2015)