PT3: A Transformer-based Model for Sepsis Death Risk Prediction via Vital Signs Time Series

1st Ruihong Luo*
School of Software
Tsinghua University
Beijing, China
lrh21@mails.tsinghua.edu.cn

2nd Minghui Gong*
School of Software
Tsinghua University
Beijing, China
gmh19@mails.tsinghua.edu.cn

3rd Chunping Li
School of Software
Tsinghua University
Beijing, China
cli@tsinghua.edu.cn

2023 International Joint Conference on Neural Networks (IJCNN) | 978-1-6654-8867-9/23/$31.00 ©2023 IEEE | DOI: 10.1109/IJCNN54540.2023.10191261

***Abstract*—Sepsis is a life-threatening systemic syndrome with a high mortality rate. It is critical for doctors to identify sepsis patients at high risk of death in real time, saving patients and reducing in-hospital mortality. However, current clinical methods use traditional machine learning models to predict the death risk of sepsis patients based on the vital signs and lab test results, which is difficult to achieve real-time prediction. In this work, we propose a novel Transformer-based model named PT3 to predict the death risk of sepsis patients within k hours ($k = 6, 24, 48$) in the future based only on the vital signs time series that can be collected in real time. In clinical settings, the collection intervals of vital signs time series are usually irregular. To address this challenge, we design a time-aware mechanism by using a time decay function to explore temporal correlations between records at different moments. We further introduce an auto-imputing mechanism to our model by using the masked prediction pre-training task. To enhance the representation learning ability, we propose a similarity prediction task, a self-supervised pre-training method, to pre-train our model with triplet-loss function. We validate the effectiveness of PT3 on two public clinical databases, MIMIC-IV and eICU. Experiments results show that our model has effective prediction performance, whose AUC on MIMIC-IV and eICU datasets achieve 0.9067 and 0.8733 respectively, especially in the next 6 hours, outperforming other state-of-the-art deep learning methods.**

***Index Terms*—sepsis, death risk, Transformer, irregular time interval, missing value**

I. Introduction

Sepsis is a syndrome of severe systemic reactions caused by infection. It is a common complication of various trauma, burns, shock, injuries and major surgical procedures, and can damage multiple organ systems and lead to death. Increased risk of sepsis is associated with older or younger age, immunosuppression, and infection with multidrug-resistant microorganisms [1]–[3]. Despite ongoing advances in diagnostic, therapeutic and monitoring methods, the morbidity and mortality of sepsis remain high. A recent retrospective analysis reported that global incidence of sepsis is estimated to be 48.9 million cases and the number of sepsis-related deaths is about 11 million [4]. Mortality in sepsis patients increases linearly with the severity of sepsis. Uncontrolled sepsis [5] exacerbates multi-organ failure, resulting in a mortality of 19.7% [4]. When septic shock occurs, the mortality rate increases to over 40% [6]. Due to the high mortality rate of sepsis, real-time identification of sepsis patients at high risk of death is critical for doctors to intervene in time to reduce in-hospital mortality.

Up to now, the clinical mainstream approach to predict the risk of death in sepsis patients is to use traditional machine learning models based on the vital signs and lab test results [7]–[13]. However, there are two main problems with these prevailing methods. First, these methods are difficult to predict in real time because they use lab test results, which need to be collected invasively and take some time to obtain available results after the blood draw. Second, these machine learning methods only use records of patient vital signs and lab test results at a single point in time, ignoring the time-series nature of vital signs data. To address these two issues, [14] proposed a model based on Temporal Convolutional Network (TCN) that only processes time series data of vital signs to predict the risk of death in sepsis patients in the next few hours, making real-time prediction of sepsis death risk becomes possible. However, the modeling method in [14] has limitations, that is, the irregularly sampled vital signs records of patients are converted into time series with equal time intervals for further analysis, resulting in the destruction of the original information in the data. In addition, the missing values are simply filled manually, and the filled results may not correctly reflect the real state of the patient, which will have a negative impact on the model training process. In summary, vital signs time series are complex and difficult to handle because they are sampled at irregular intervals and contain a large number of missing values. This paper aims to address the challenging problem of processing patient vital signs time series data with irregular time intervals and missing values in order to improve the predictive performance of mortality risk in sepsis patients.

To address this issue, we propose a Transformer-based model, called Pre-trained Triplet-loss Time-aware Transformer (PT3), that can handle time series data with both irregular time intervals and missing values. Specifically, we design a time-aware mechanism for Transformer by using a time decay function, which can capture time information into PT3, enabling the model to explore the temporal correlations between vital signs records at different moments. We further propose a pre-training task for masked prediction to pre-train the model to learn to automatically impute the missing values,

* Equal contribution

called the auto-imputing mechanism. Due to the limitation of collecting labeled medical data, it is difficult to learn effective representations from a small amount of data in a supervised manner. To enhance representation learning ability, we propose a self-supervised pre-training method for similarity prediction to pre-train the model PT3 with triplet-loss function. The contributions of our work are:

1) We propose a Transformer-based model that can simultaneously handle time series with irregular time intervals and missing values by introducing time-aware mechanism and auto-imputing mechanism.
2) A new pre-training method is designed to enhance the representation learning ability for time series data with triplet-loss function.
3) Experiments on two medical databases, i.e., MIMIC-IV and eICU, verify the effectiveness of the proposed model for warning the death risk of sepsis patients in the next k hours ($k = 6, 24, 48$) based on only vital signs.

II. Related Works

RNN-based models, such as Long Short-Term Memory (LSTM) [15] and Gated Recurrent Unit (GRU) [16], are the most classical and widely used models for processing time series data. T-LSTM [17] and GRU-D [18] can process time series data with irregular time intervals by introducing a time decay function based on LSTM and GRU respectively. Meanwhile, GRU-D can automatically predict and impute missing values.

With the widespread application of Transformer-based models [19], [20] in the field of NLP, some Transformer-based works have been proposed and applied to time series processing task. [21] only uses Transformer decoder to predict how the data will change in the future based on the historical information of the time series. Experimental comparisons with other deep learning models such as LSTMs show that Transformer performs best for time series data. LogSparse Transformer [22] and Informer [23] improve the classic Transformer structure, reduce the time and space complexity, break the bottleneck of memory usage, and thus solve the long-term prediction problem of time series. However, these models cannot handle time series data with irregular time intervals or missing values.

Some Transformer-based models are proposed for dealing with time series data with irregular time intervals or with missing values separately, achieving better performance than RNN-based models. HiTANet proposed in [24] regards patient visits as irregular sampled time series data, and use time-aware Transformer and time-aware key-query attention mechanism to model local and global temporal information for predicting patient future disease risk. [25] proposed a generalized representation learning method named RAPT, that can handle time series data with irregular time intervals by encoding the temporal information and using time-aware self-attention. RAPT is pre-trained using three tasks in a self-supervised manner, including the masked prediction, the similarity prediction and the reasonability check. However, the masked prediction task in RAPT cannot solve the problem of missing values in time series data, and the similarity between two time series in the similarity prediction task is only measured according to the last record, which is not suitable for time series data. To predict and impute missing values automatically, [26] proposes a Transformer-based framework for multivariate time series representation learning, which pre-trains the model by restoring a part of randomly masked input data, and fine-tunes the model on the dataset of the downstream task. However, to our best knowledge, currently there is no Transformer-based model that can process time series data with irregular time intervals and missing values at the same time.

III. Preliminaries

A. Problem statement

We define the sepsis death risk prediction problem as predicting the risk of death in the next k hours ($k = 6, 24, 48$) of a sepsis patient given the vital signs records in the past 24 hours. The vital signs we concern about in this work include heart rate (HR), respiratory rate (RR), oximetry saturation (SpO_2), and the mean arterial pressure (MAP), which are closely related to the progression of sepsis and easy to collect. We extract sepsis patients from MIMIC-IV [27] and eICU [28] database following the method described in [29], and then collect four vital signs records from their ICU admission to discharge or death. We regard HR records out of the range [30, 260], RR records out of the range [5, 70], SpO_2 records out of the range [0, 100], and MAP records out of the range [10, 200] as error records and delete them.

B. Data notation

For a sepsis patient, his vital signs sequential record is denoted as $\mathbf{S} =< \mathbf{X}, \mathbf{M}, \mathbf{T} >$. $\mathbf{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_T\}^\top \in \mathbb{R}^{T\times 4}$ is the time series data of four vital signs in the past 24 hours, where T is the sequence length and $\boldsymbol{x}_i \in \mathbb{R}^4 (1 \leq i \leq T)$ is the vector consisting of four vital signs in the ith record. x_{ij} is the value of vital sign j in $\boldsymbol{x}_i$. $\mathbf{M} = \{\boldsymbol{m}_1, \boldsymbol{m}_2, \cdots, \boldsymbol{m}_T\}^\top \in \mathbb{R}^{T\times 4}$ is the missing mask matrix generated from $\mathbf{X}$. When x_{ij} is missing, m_{ij} is 0, otherwise m_{ij} is 1. Suppose the recording time of the ith record is noted as t_i, then $\mathbf{T} = \{t_T - t_1, t_T - t_2, \cdots, t_T - t_{T-1}, 0\}$ denotes the time information vector of the patient, which is used to mark the hourly interval between each record and the last record.

C. Pre-training dataset

The pre-training dataset is used to pre-train PT3 through the masked prediction task and the similarity prediction task. The pre-training dataset is constructed from the MIMIC-IV database. For each patient, we collect instances using a 24-hour sliding window from the ICU admission until the window starts later than the moment of ICU discharge or death, as shown in Fig. 1. The one-hour interval between two adjacent windows is to ensure that there is no time overlap between two adjacent time series. Finally, there are 71,826 instances in the pre-training dateset.

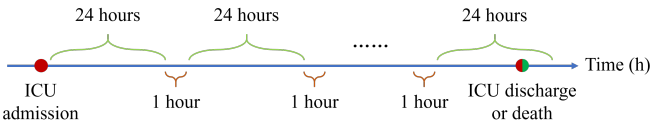


Fig. 1: Conceptual illustration of extracting instances for the pre-training dataset.

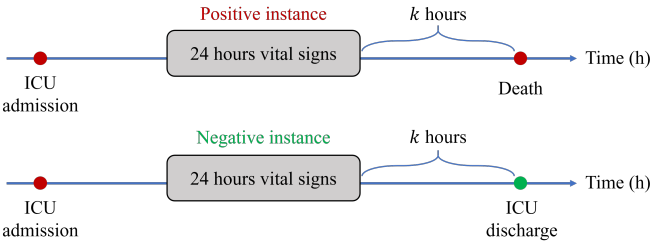


Fig. 2: Conceptual illustration of extracting instances for the prediction dataset.

D. Prediction dataset

The prediction dataset is used to fine-tune our pre-trained model to predict the death risk in sepsis patients in the next k hours, where $k = 6, 24, 48$, according to the vital signs in the past 24 hours. For each k, we construct two binary classification datasets from MIMIC-IV and eICU database respectively. Vital signs are extracted from the 24-hour window of k hours before the death of the deceased patient as a positive instance, and from the 24-hour window of k hours before the ICU discharge of the surviving patient as a negative instance, as shown in Fig. 2. The number of positive and negative instances in different datasets is shown in Table I. The MIMIC-IV dataset is devided into training dataset, validation dataset and test dataset in the ratio of $7:1:2$, which are used for training, validation and testing of the PT3 respectively. The datasets from eICU are used as external validation dataset to verify the predictive performance of the model on new data.

IV. Methodology

A. Model structure

The structure of PT3 is similar to the Transformer encoder, as shown in Fig. 3. Given a vital signs sequential record $\mathbf{S} =< \mathbf{X}, \mathbf{M}, \mathbf{T} >$, we first encode them to a high-dimensional space using a fully connected layer and $tanh$ activation function

TABLE I: The number of instances in the prediction dataset.

k	MIMIC-IV		eICU	
	Positive	Negative	Positive	Negative
6	1624	13245	1821	15198
24	1406	10463	1292	12070
48	1169	7118	962	8458

to obtain the vital signs embedding $\mathbf{E}^x$, the missing mask embedding $\mathbf{E}^m$ and the temporal embedding $\mathbf{E}^\tau$ as:

$$\mathbf{E}^x = \tanh\left(\mathbf{X}\mathbf{W}_x + \boldsymbol{b}_x\right) \tag{1}$$

$$\mathbf{E}^m = \tanh\left(\mathbf{M}\mathbf{W}_m + \boldsymbol{b}_m\right) \tag{2}$$

$$\mathbf{E}^\tau = \tanh\left(\mathbf{T}\mathbf{W}_\tau + \boldsymbol{b}_\tau\right) \tag{3}$$

where $\mathbf{W}_x, \mathbf{W}_m \in \mathbb{R}^{4\times d_{model}}$, $\mathbf{W}_\tau \in \mathbb{R}^{1\times d_{model}}$ and $\boldsymbol{b}_x, \boldsymbol{b}_m, \boldsymbol{b}_\tau \in \mathbb{R}^{d_{model}}$ are learnable parameters. Then, we obtain the input embedding $\mathbf{E}$ by summing $\mathbf{E}^x$ and $\mathbf{E}^m$ as:

$$\mathbf{E} = \mathbf{E}^x + \mathbf{E}^m \tag{4}$$

Suppose there are n heads in the multi-head attention layer. In the nth layer, we compute the attention score matrix $\mathbf{A}^{(n)}$ using the input embedding $\mathbf{E}$ and the temporal embedding $\mathbf{E}^\tau$. First, we get $\mathbf{Q}^{(n)}$, $\mathbf{K}^{(n)}$ and $\mathbf{V}^{(n)}$ as:

$$\mathbf{Q}^{(n)} = \mathbf{E}\mathbf{W}_q^{(n)} + \boldsymbol{b}_q^{(n)} \tag{5}$$

$$\mathbf{K}^{(n)} = \mathbf{E}\mathbf{W}_k^{(n)} + \boldsymbol{b}_k^{(n)} \tag{6}$$

$$\mathbf{V}^{(n)} = \mathbf{E}\mathbf{W}_v^{(n)} + \boldsymbol{b}_v^{(n)} \tag{7}$$

where $\mathbf{W}_q^{(n)}, \mathbf{W}_k^{(n)}, \mathbf{W}_v^{(n)} \in \mathbb{R}^{d_{model}\times d_h}$ and $\boldsymbol{b}_q^{(n)}, \boldsymbol{b}_k^{(n)}, \boldsymbol{b}_v^{(n)} \in \mathbb{R}^{d_h}$ are learnable parameters, and $d_h = d_{model}/N$. To enable the model to handle time series data with irregular time intervals, when calculating each element $a_{ij}^{(n)}$ in $\mathbf{A}^{(n)}$, we design the time-aware mechanism by acting the time decay function $g(\Delta) = 1/\log(e+\Delta)$ on the time interval of the ith record and the jth record to obtain the temporal information and introduce it into the attention score:

$$a_{ij}^{(n)} = \frac{\boldsymbol{q}_i^{(n)} \cdot \boldsymbol{k}_j^{(n)}}{\sqrt{d_h}} + \frac{1}{\log\left(e + \left|\mathbf{W}_t^{(n)} | \boldsymbol{e}_i^\tau - \boldsymbol{e}_j^\tau |\right|\right)} \tag{8}$$

where $\mathbf{W}_t^{(n)} \in \mathbb{R}^{d_{model}\times 1}$ is learnable parameters, and $|\cdot|$ is elementwise absolute value. The time decay function decreases monotonically as the time interval increases, making the farther moments have less effect on the current moment. Then, we can get the output of the nth head $\mathbf{H}^{(n)}$ after calculating the attention score matrix $\mathbf{A}^{(n)}$:

$$\mathbf{H}^{(\mathbf{n})} = Softmax\left(\mathbf{A}^{(n)}\right)\mathbf{V}^{(n)} \tag{9}$$

Finally, the output of all N heads is concatenated together and passed through a fully connected layer to obtain the output of the multi-head attention layer:

$$\mathbf{H} = Concat\left(\mathbf{H}^{(1)}, \mathbf{H}^{(2)}, \cdots, \mathbf{H}^{(N)}\right)\mathbf{W}_o + \boldsymbol{b}_o \tag{10}$$

where $\mathbf{W}_o \in \mathbb{R}^{d_{model}\times d_{model}}$ and $b_o \in \mathbb{R}^{d_{model}}$ are learnable parameters.

The multi-head attention layer is followed by a residual connection. We adopt the suggestion in [26] to replace the layer normalization in the original Transformer encoder with batch normalization which can mitigate the effect of outliers in time series data. Then, a feed-forward layer is used to process the output of the first batch normalization layer $\mathbf{E}^{BN}$ as:

$$FFN(\mathbf{E}^{BN}) = ReLU\left(\mathbf{E}^{BN}\mathbf{W}_1 + \boldsymbol{b}_1\right)\mathbf{W}_2 + \boldsymbol{b}_2 \tag{11}$$

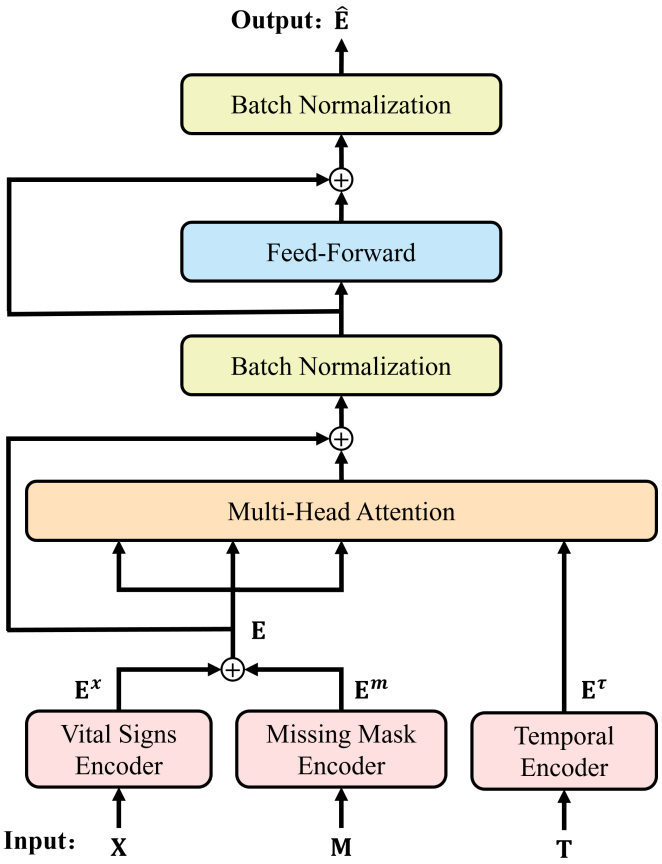


Fig. 3: The structure of PT3.

where $\mathbf{W}_1 \in \mathbb{R}^{d_{model} \times d_{ff}}$, $\mathbf{W}_2 \in \mathbb{R}^{d_{ff} \times d_{model}}$, $\boldsymbol{b}_1 \in \mathbb{R}^{b_{ff}}$ and $\boldsymbol{b}_2 \in \mathbb{R}^{b_{model}}$ are learnable parameters. After another residual connection and batch normalization layer, we get the output of PT3, denoted as $\hat{\mathbf{E}}$. The vector at the last time step of $\hat{\mathbf{E}}$, denoted as $\hat{\boldsymbol{e}}_T$, is regarded as the representation vector of the patient's vital signs sequential record in the past 24 hours.

B. Pre-training process

In this section, we introduce two pre-training tasks and explain how to pre-train the model PT3 with these two tasks at the same time.

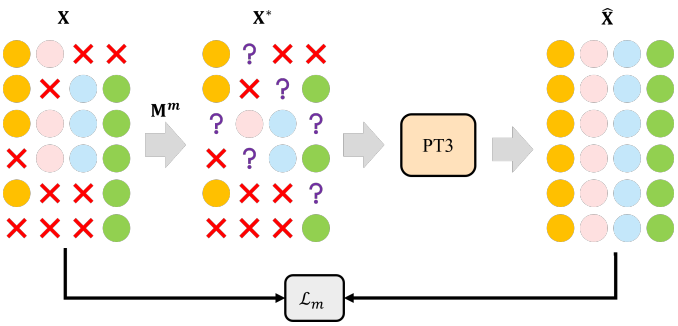


Fig. 4: The process of the masked prediction task.

1) Masked prediction: Due to reasons such as storage lost or recording errors, there are a large number of missing values in patients vital signs records. However, manual methods of filling missing values, such as forward filling or average filling, cannot reflect the real state of the patient, which may bring negative impact on the model training process. Therefore, we design an auto-imputing mechanism through the masked prediction task to pre-train PT3, which enables the model to automatically impute the missing values.

The training process of the masked prediction task is to randomly mask some non-missing values, and let the model learn how to predict and restore them based on the other non-missing values that have not been masked, as shown in Fig. 4, where red crosses represent missing values and purple question marks represent masked records. Given a vital signs time series $\mathbf{S} =< \mathbf{X}, \mathbf{M}, \mathbf{T} >$, we create a random masked matrix $\mathbf{M}^m$ with the same dimensions as $\mathbf{M}$. First, we set masked ratio as $p_m \in (0, 1)$, and randomly let the elements in $\mathbf{M}^m$ with a proportion of p_m be 0 and others be 1. Then, we get the new time series data $\mathbf{X}^*$ and missing mask matrix $\mathbf{M}^*$ as:

$$\mathbf{X}^* = \mathbf{X} \odot (1 - \mathbf{M}^m) \tag{12}$$

$$\mathbf{M}^* = \mathbf{M} \odot (1 - \mathbf{M}^m) \tag{13}$$

where $\odot$ is elementwise multiplication. After imputing the new record $< \mathbf{X}^*, \mathbf{M}^*, \mathbf{T} >$ into PT3 and getting the output $\hat{\mathbf{E}}$, we restore the vital signs record as:

$$\hat{\mathbf{X}} = \hat{\mathbf{E}}\mathbf{W}_e + \boldsymbol{b}_e \tag{14}$$

where $\mathbf{W}_e \in \mathbb{R}^{d_{model} \times 4}$ and $b_e \in \mathbb{R}^4$ are learnable parameters. We use mean square error (MSE) as the loss function of the masked prediction task:

$$\mathcal{L}_m = \frac{1}{|\hat{\mathbf{M}}|} \sum_{(i,j) \in \hat{\mathbf{M}}} (x_{ij} - \hat{x}_{ij})^2 \tag{15}$$

where $\hat{\mathbf{M}} = \{(i, j) | \mathbf{M}_{ij} = 1, \mathbf{M}^m_{ij} = 1\}$ is the set consisting of the positions of the masked non-missing values, and $|\hat{\mathbf{M}}|$ is the number of elements in $\hat{\mathbf{M}}$. $\mathbf{M}_{ij}$ and $\mathbf{M}^m_{ij}$ are the elements located in the ith row and jth column of the matrices $\mathbf{M}$ and $\mathbf{M}^m$, respectively.

2) Similarity prediction: Our proposed PT3 can learn representations of patient's vital signs records in the past 24 hours. If the model is able to learn similar representations for patients with similar vital sign changes or disease progression, and different representations for patients with large differences, the model can better distinguish patients with different health conditions, so that the learned representation will be more informative for downstream tasks. To achieve this goal, we implement the similarity prediction task to perform self-supervised pre-training of PT3 to enhance representation learning for time series data.

Before this task, we need to identify a Similar Group and a Different Group for each instance in the pre-training dataset. First, we use the Dynamic Time Warping (DTW) algorithm [30] to measure the similarity of each instance pair, which can also be viewed as the DTW distance between two instances.

The smaller the distance between two instances, the more similar they are. Then, we set similarity ratio as $p_s \in (0, 0.5]$. The largest value among the top p_s proportionally small DTW distance values is used as the similarity distance threshold d_{sim}, and the smallest value among the top p_s proportionally large DTW distance values is used as the difference distance threshold d_{diff}, as shown in Fig. 5. For any instance $\mathbf{S}$, its Similar Group consists of all instances whose DTW distance to $\mathbf{S}$ is no greater than d_{sim}, and its Different Group consists of all instances whose DTW distance to $\mathbf{S}$ is no less than d_{diff}.

The training process for the similarity prediction task is shown in Fig. 6. For each instance $\mathbf{S}$, in each epoch, we randomly select a similar instance $\mathbf{S}^{pos}$ and a different instance $\mathbf{S}^{neg}$ from the Similar Group and the Different Group of $\mathbf{S}$, respectively, and obtain their representation vectors $\hat{\boldsymbol{e}}$, $\hat{\boldsymbol{e}}^{pos}$ and $\hat{\boldsymbol{e}}^{neg}$. Then, we use the triplet loss that is used in [31] as the loss function of the similarity prediction task:

$$\mathcal{L}_s = -\log\left(\sigma(\hat{\boldsymbol{e}}^{pos}\hat{\boldsymbol{e}}^{\top})\right) - \log\left(\sigma(-\hat{\boldsymbol{e}}^{neg}\hat{\boldsymbol{e}}^{\top})\right) \tag{16}$$

where σ is the sigmoid activation function. This loss pushes the computed representation vectors to assimilate $\mathbf{S}$ and $\mathbf{S}^{pos}$, and to distinguish between $\mathbf{S}$ and $\mathbf{S}^{neg}$.

3) Pre-training Loss: During the pre-training phase, we train the model by performing the masked prediction task and the similarity prediction task at the same time, and the final pre-training loss is defined as:

$$\mathcal{L}_{pretrain} = \lambda\mathcal{L}_m + (1-\lambda)\mathcal{L}_s \tag{17}$$

where λ is a hyperparameter to balance the masked prediction task and the similarity prediction task.

C. Death risk prediction

After pre-training, we initialize the model with the pre-trained parameters and fine-tune it with the prediction dataset. For an instance $\mathbf{S} =< \mathbf{X}, \mathbf{M}, \mathbf{T} >$ of a patient, we get its representation vector $\hat{\boldsymbol{e}}$ first, and then feed it into a new fully-connected layer and a sigmoid activation function sequentially to obtain the death risk in the next k hours for this patient, which is denoted as $\hat{y} \in (0, 1)$. For a dataset consisting of P patients, $\mathcal{D} = \{(\mathbf{X}^p, \mathbf{M}^p, \mathbf{T}^p, y^p)|p = 1, 2, \cdots, P\}$, we use the binary cross entropy as the cost function as follow:

$$\mathcal{L}(\mathcal{D}; \mathbf{w}) = -\frac{1}{P}\sum_{p}^{P} y^p \log\hat{y} + (1-y^p)\log\left(1-\hat{y}\right) \tag{18}$$

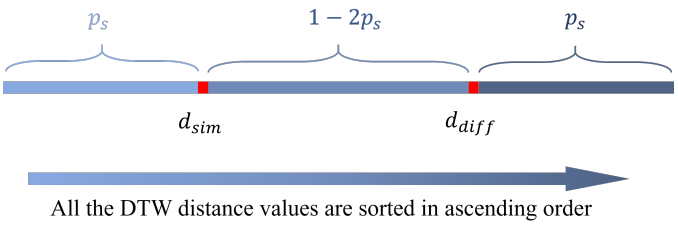


Fig. 5: Method of determining two distance thresholds.

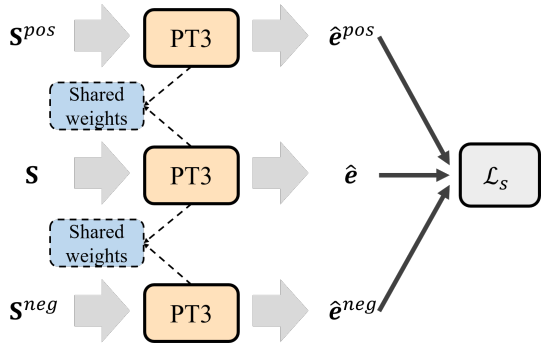


Fig. 6: The process of the similarity prediction task.

V. Experiments

In this section, we first introduce the experimental settings and then analyze the experimental results to demonstrate the effectiveness of PT3.

A. Experimental setup

1) Datasets: We conduct experiments on two publicly available medical databases, MIMIC-IV and eICU. MIMIC-IV is a large single-center database that collects comprehensive information of patients admitted to the Beth Israel Deaconess Medical Center from 2008 to 2019, including lab test results, drug use and vital signs time series. eICU is a multi-center database comprising heath record data from more than 200,000 ICU patients in United States between 2014 and 2015. These two databases have different structures and contents, and we screen out sepsis patient records to construct the experimental datasets. In our experiments, we train models on MIMIC-IV and evaluate models on both MIMIC-IV and eICU to verify the effectiveness and generalization of the proposed model.

2) Baseline methods: In our work, we evaluate the model PT3 against the following state-of-the-art models that all can process time series data with irregular time intervals.

- T-LSTM [17]: An RNN-based model that can process time series data with irregular time intervals by introducing time decay function into the LSTM cell.
- GRU-D [18]: An RNN-based model that can process time series data with irregular time intervals using learnable decay function, as well as imputing missing values automatically at the same time.
- RAPT [25]: A Transformer-based model that can process time series data with irregular time intervals by introducing time-aware self-attention mechanism.

3) Evaluation metrics: We compare the predictive performance of each model using the Area Under the ROC Curve (AUC), the Area Under Precision-Recall Curve (AUPRC), and three other metrics that are commonly used for binary classification problems in the medical field:

- Sensitivity (Sen): Equals to the True Positive Rate (TPR), which reflects the accuracy of the model for identifying patients with high death risk.

TABLE II: The experiment results on sepsis patient death risk prediction. Bold/underline indicates the best/second.

	Model	MIMIC-IV					eICU				
		Sen	Spe	Youden	AUC	AUPRC	Sen	Spe	Youden	AUC	AUPRC
$k=6$	T-LSTM	0.7729	0.8443	0.6172	0.8866	0.6829	<u>0.7779</u>	0.7444	0.5223	0.8426	0.5004
	GRU-D	0.7828	0.8547	0.6374	0.8940	**0.7054**	**0.8298**	0.6893	0.5191	0.8555	**0.5799**
	RAPT	<u>0.7871</u>	<u>0.8564</u>	<u>0.6435</u>	<u>0.8960</u>	<u>0.6972</u>	0.7778	<u>0.7854</u>	<u>0.5633</u>	<u>0.8579</u>	0.4898
	PT3	**0.8052**	**0.8612**	**0.6636**	**0.9067**	0.6961	0.7660	**0.8186**	**0.5845**	**0.8733**	<u>0.5798</u>
$k=24$	T-LSTM	0.5965	<u>0.7714</u>	0.3679	0.7479	0.3944	0.5864	**0.7933**	0.3796	0.7359	0.3121
	GRU-D	0.6291	**0.7785**	0.4076	0.7724	<u>0.4236</u>	<u>0.6596</u>	0.7383	<u>0.3979</u>	<u>0.7646</u>	<u>0.3180</u>
	RAPT	**0.6787**	0.7532	**0.4319**	<u>0.7776</u>	0.3943	0.6050	<u>0.7868</u>	0.3918	0.7514	0.2826
	PT3	<u>0.6681</u>	0.7438	<u>0.4119</u>	**0.7841**	**0.4374**	**0.6780**	0.7289	**0.4069**	**0.7755**	**0.3744**
$k=48$	T-LSTM	0.5632	<u>0.6912</u>	0.2544	0.6799	0.3147	0.4719	**0.7906**	0.2626	0.6774	<u>0.2264</u>
	GRU-D	<u>0.5906</u>	0.6805	**0.2711**	<u>0.6896</u>	<u>0.3242</u>	<u>0.5437</u>	0.7299	**0.2736**	<u>0.6825</u>	0.2024
	RAPT	0.5641	**0.6968**	0.2609	0.6746	0.2866	0.5071	<u>0.7501</u>	0.2572	0.6784	0.2087
	PT3	**0.6154**	0.6506	<u>0.2659</u>	**0.7055**	**0.3340**	**0.6609**	0.6102	<u>0.2711</u>	**0.6941**	**0.2653**

- Specificity (Spe): Equals to the True Negative Rate (TNR), which reflects the accuracy of the model for identifying patients with low death risk.
- Youden Index (Youden): Equals to $Sen+Spe-1$, which reflects the accuracy of the model for identifying patients (positive) and non-patients (negative).

We expect PT3 to be able to accurately distinguish patients with high risk of death from those with low risk of death, and identify sepsis patients with high death risk as accurately as possible. Therefore, the results of Youden Index, AUC and Sensitivity are more important than Specificity in this work. Since all datasets are imbalanced, we do not use Accuracy as an evaluation metric, but use the AUPRC instead.

4) Implementation details: We implement PT3 and all the baselines in PyTorch 1.10.2 and train them on Ubuntu 20.04 with 128GB memory and a GeForce GTX 1080 Ti GPU. The hyperparameters are set according to the base model of Transformer in [19], where $d_{model} = 512, N = 8, d_h = 64, d_{ff} = 2048$. For hyperparameters in the pre-training process, we set $p_m = 0.3, p_s = 0.3, \lambda = 0.6$. We run each model five times and compare the average of each metric.

B. Experiment results and analysis

Table II shows the comparison of the prediction performance of different models on MIMIC-IV and eICU. The optimal and suboptimal results for a metric under the same dataset are marked in bold and underline respectively. By comparing the experimental results, we can see that among all 24 results for the four metrics of Youden Index, AUC, AUPRC and Sensitivity, 22 results of PT3 rank in the top two, accounting for 91.7%, and 17 results rank the first place, accounting for 70.8%, which indicates that the representations learned by PT3 are more effective in distinguishing sepsis patients with high death risk from those with low death risk, and can timely and accurately identify sepsis patients with high death risk. Therefore, PT3 has the best prediction performance among all the models.

Further analyzing the experimental results, we observe that for each model, the prediction effect decreases as the value of k increases, because the physiological state of the patient changes every moment, making it more uncertain to predict a more distant state based on the patient's current condition. Therefore, the problem of predicting the risk of death from sepsis becomes difficult as the prediction timeline increases. On these two datasets with $k = 6$, the two Transformer-based pre-trained models PT3 and RAPT outperform two RNN-based models. However, on both datasets with $k = 48$, PT3 and GRU-D, which can automatically predict and impute missing values, achieve better prediction results than RAPT and T-LSTM, which need to manually fill in missing values. This indicates that missing values have less negative impact on the model when the prediction problem is not very difficult. At this time, using a large amount of data to pre-train the model can improve the representation learning ability of the model, so PT3 and RAPT can achieve better prediction performance. As the difficulty of the prediction problem increases, missing values can bring more negative effects on the training and prediction process, reducing the representation learning ability of the model. The missing data filled manually are simply estimated values, which cannot reflect the real state of the patient at that time. Therefore, introducing the auto-imputing mechanism can mitigate the negative impact of missing values, enhance the representation learning ability of the model, and improve the prediction performance.

Although our work mainly focuses on the death risk of sepsis patients in the next 48 hours, if trained on a longer time series, PT3 can predict the death risk on a longer time scale. Additionally, based on the Transformer structure, PT3 breaks the serial structure of the RNN-based model with stronger parallelism and shorter training time. When applied in clinical settings, the model can be trained offline in advance, and then inferred in real time to obtain better prediction performance.

C. Ablation study

PT3, pre-trained by the masked prediction task and the similarity prediction task, introduce a time-aware mechanism to handle irregular time interval by using a time decay function. Moreover, batch normalization is used instead of layer

TABLE III: Ablation study results ($k = 6$).

	Youden			AUC		
	MIMIC-IV	eICU	Avg.	MIMIC-IV	eICU	Avg.
PT3 w/o pt	0.6125	0.5564	0.5845	0.8885	0.8564	0.8725
PT3 w/o ta	0.6399	0.5605	0.6002	0.8978	0.8642	0.8810
PT3 w/o mp	0.6535	0.5740	0.6138	0.9023	0.8680	0.8852
PT3 w/o sp	0.6445	**0.5860**	0.6153	0.9039	0.8708	0.8874
PT3 w/o bn	**0.6677**	0.5697	0.6187	0.9059	0.8705	0.8882
PT3	0.6636	0.5845	**0.6241**	**0.9067**	**0.8733**	**0.8900**

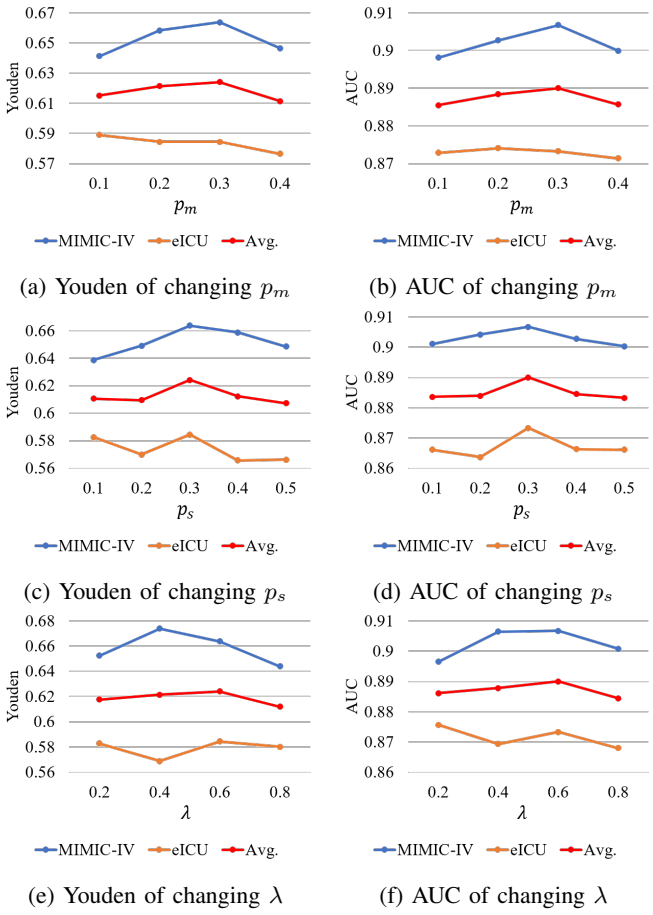
normalization to mitigate the effect of outliers in time series data. To determine the actual contribution of each part of the model setup to the final performance, we report the results of an ablation study on the datasets with $k = 6$ in Table III. The evaluation metrics used here are the mean of the Youden Index and AUC on MIMIC-IV and eICU datasets. We compare PT3 with the following variants:

- PT3 w/o pt: PT3 trained end-to-end and without two pre-training tasks.
- PT3 w/o ta: PT3 without time-aware mechanism. The temporal embedding $\mathbf{E}^{\tau}$ is added to the input embedding $\mathbf{E}$.
- PT3 w/o mp: PT3 without the masked prediction task.
- PT3 w/o sp: PT3 without the similarity prediction task.
- PT3 w/o bn: PT3 that use layer normalization instead of batch normalization.

Among all the variants, PT3 w/o pt performs worst, which indicates that pre-training can substantially improve the prediction performance of the model, and both the masked prediction task and the similarity prediction task are helpful. The models with the second and third worst prediction performance are PT3 w/o ta and PT3 w/o mp, respectively. This indicates that when processing time series data with irregular time intervals and missing values, we should focus on how to deal with the irregular temporal information of the data as well as the missing values to avoid their negative impact on the model training process as much as possible. In addition, using layer normalization instead of batch normalization slightly improves the Youden Index on the MIMIC-IV dataset but substantially decreases it on the eICU dataset, which means batch normalization is more suitable than layer normalization for processing the vital signs time series data and can improve the generalization ability of the model.

D. Hyperparameters setting

There are three hyperparameters used in the pre-training process, which are the masked ratio p_m, the similarity ratio p_s and the loss function weight λ in (17). To determine the optimal pre-training hyperparameters combination, we vary one hyperparameter at a time and compare the average score of Youden Index and AUC of PT3 on MIMIC-IV and eICU with $k = 6$. As shown in Fig. 7, $p_m = 0.3$, $p_s = 0.3$ and $\lambda = 0.6$ lead to the highest average Youden Index and AUC.



(a) Youden of changing p_m (b) AUC of changing p_m

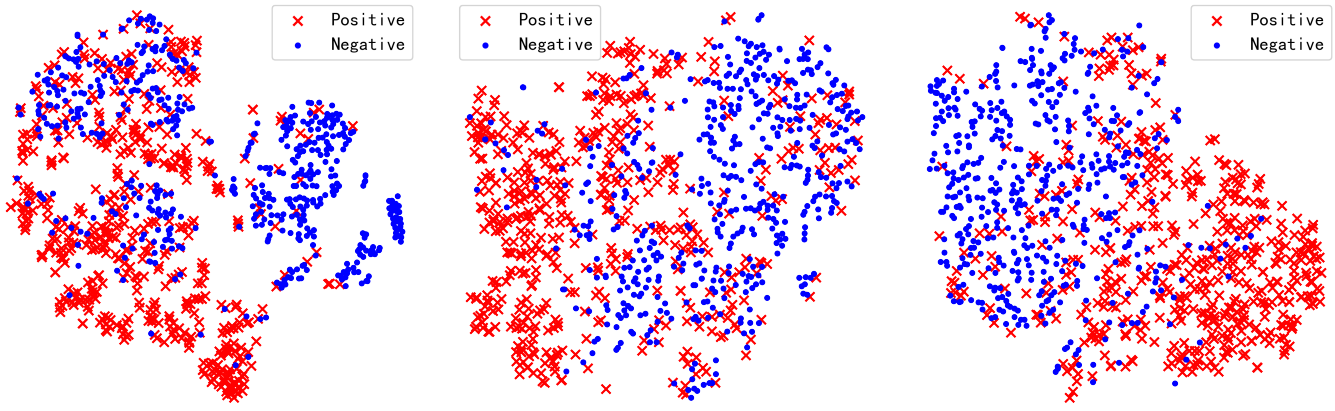
(c) Youden of changing p_s (d) AUC of changing p_s

(e) Youden of changing λ (f) AUC of changing λ

Fig. 7: Performance comparison with changing hyperparameters ($k = 6$).

E. Representations visualization

To illustrate that PT3 can learn effective representations from patient vital signs time series data, we randomly select 500 positive and 500 negative instances from the eICU dataset with $k = 6$, and obtain the representation vectors of these 1000 instances respectively from PT3, PT3 w/o pt and a pre-trained PT3 without fine-tuning called PT3 w/o ft. We then visualize the representation vectors learned by these models using t-SNE [32], as shown in Fig. 8. Among all the results, the learned representations of PT3 w/o pt have the worst performance in distinguishing positive from negative instances. PT3 performs



(a) PT3 w/o pt (b) PT3 w/o ft (c) PT3

Fig. 8: Visualization of the representations learned by different models.

the best, as the positive instances are mainly distributed in the lower right part of Fig. 8c, while the negative instances are mainly distributed in the upper left part, and we can distinguish most of the positive instances from the negative instances by a straight line. After comparison, it can be concluded that both the pre-training process and the fine-tuning process are helpful to enhance the representation learning ability of the model, and PT3 is able to learn effective representations from the patient's vital signs time series data.

VI. Conclusion and Future Works

In our work, we propose a Transformer-based model, named PT3, that can simultaneously handle time series data with irregular time intervals and missing values. By introducing a time decay function, we design a time-aware mechanism for PT3 model, enabling it to explore temporal correlations between vital signs records at different moments. To enable the model to automatically impute missing values, we further design an auto-imputing mechanism for PT3, i.e., we pre-train the model by constructing a masked prediction task. In addition, to enhance the representation learning ability, we pre-train PT3 with a similarity prediction task, which is a self-supervised pre-training method. The experimental results on death risk prediction in sepsis patients show that PT3 has effective prediction performance, outperforming other state-of-the-art deep learning methods. Our future objective is to further broaden the application of the model to predict the risk of death in ICU patients in real time, not limited to sepsis patients.

References

[1] J. Blanco, A. Muriel-Bombin, V. Sagredo, F. Taboada, F. Gandia, and L. Tamayo et al., "Incidence, organ dysfunction and mortality in severe sepsis: a Spanish multicentre study," in *Critical Care*, vol. 12, no. 6, 2008.

[2] A. M. Esper and G. S. Martin, "Extending international sepsis epidemiology: the impact of organ dysfunction," in *Critical Care*, vol. 13, no. 1, 2009.

[3] K. M. Kaukonen, M. Bailey, S. Suzuki, D. Pilcher, and R. Bellomo, "Mortality related to severe sepsis and septic shock among critically ill patients in Australia and New Zealand," in *JAMA*, vol. 311, no. 13, pp. 1308-1316, 2014.

[4] K. E. Rudd, S. C. Johnson, K. M. Agesa, K. A. Shackelford, D. Tsoi, and K. D. Derrick et al., "Global, regional, and national sepsis incidence and mortality, 1990–2017: analysis for the Global Burden of Disease Study," in *The Lancet*, vol. 395, no. 10219, pp. 200-211, 2020.

[5] S. M. Perman, M. Goyal, and D. F. Gaieski, "Initial emergency department diagnosis and management of adult patients with severe sepsis and septic shock," in *Scandinavian Journal of Trauma, Resuscitation and Emergency Medicine*, vol. 20, no. 1, 2012.

[6] M. S. Rangel-Frausto, D. Pittet, M. Costigan, T. Hwang, C. S. Davis, and R. P. Wenzel, "The natural history of the systemic inflammatory response syndrome (SIRS): a prospective study," in *JAMA*, vol. 273, no. 2, pp. 117-123, 1995.

[7] R. A. Taylor, J. R. Pare, A. K. Venkatesh, H. Mowafi, E. R. Melnick, and W. Fleischman et al., "Prediction of in-hospital mortality in emergency department patients with sepsis: a local big data–driven, machine learning approach," in *Academic Emergency Medicine*, vol. 23, no. 3, pp. 269-278, 2016.

[8] A. Khojandi, V. Tansakul, X. Li, R. S. Koszalinski, and W. Paiva, "Prediction of sepsis and in-hospital mortality using electronic health records," in *Methods of Information in Medicine*, vol. 57, no. 4, pp. 185-193, 2018.

[9] A. Rodriguez, D. Mendoza, J. Ascuntar, and F. Jaimes, "Supervised classification techniques for prediction of mortality in adult patients with sepsis," in *The American Journal of Emergency Medicine*, vol. 45, pp. 392-397, 2021.

[10] G. Kong, K. Lin, and Y. Hu, "Using machine learning methods to predict in-hospital mortality of sepsis patients in the ICU," in *BMC Medical Informatics and Decision Making*, vol. 20, no. 1, pp. 251-260, 2020.

[11] N. Hou, M. Li, L. He, B. Xie, L. Wang, and R. Zhang et al., "Predicting 30-days mortality for MIMIC-III patients with sepsis-3: a machine learning approach using XGboost," in *Journal of Translational Medicine*, vol. 18, no. 1, 2020.

[12] W. van Doorn, P. Stassen, H. Borggreve, M. Schalkwijk, J. Stoffers, and O. Bekers et al., "A comparison of machine learning models versus clinical evaluation for mortality prediction in patients with sepsis," in *PloS One*, vol. 16, no. 1, 2021.

[13] J. Perng, I. Kao, C. Kung, S. Hung, Y. Lai, and C. Su, "Mortality prediction of septic patients in the emergency department based on machine learning," in *Journal of Clinical Medicine*, vol. 8, no. 11, pp. 1906-1922, 2019.

[14] M. Gong, J. Liu, C. Li, W. Guo, R. Wang, and Z. Chen, "Early warning model for death of sepsis via length insensitive temporal convolutional network," in *MBEC*, vol. 60, no. 3, pp. 875-885, 2022.

[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," in *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[16] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, and H. Schwenk, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *EMNLP*, pp. 1724-1734, 2014.
[17] I. M. Baytas, C. Xiao, X. Zhang, F. Wang, and A. K. Jain, "Patient subtyping via time-aware LSTM networks," in *SIGKDD*, pp. 65-74, 2017.
[18] Z. Che, S. Purushotham, K. Cho, D. Sontag, and Y. Liu, "Recurrent neural networks for multivariate time series with missing values." in *Scientific Reports*, vol. 8, no. 1, 2018.
[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, and A. N. Gomez et al., "Attention is all you need," in *NeurIPS*, pp. 5998-6008, 2017.
[20] J. Devlin, M. W. Chang, K. Lee and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *NAACL*, pp. 4171-4186, 2019.
[21] P. Lara-Benítez, L. Gallego-Ledesma, M. Carranza-García, and J. M. Luna-Romera, "Evaluation of the transformer architecture for univariate time series forecasting," in *Conference of the Spanish Association for Artificial Intelligence*, pp. 106-115, 2021.
[22] S. Li, X. Jin, Y. Xuan, X. Zhou, W. Chen, and Y. X. Wang et al., "Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting," in *NeurIPS*, pp. 5244-5254, 2019.
[23] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, and H. Xiong et al., "Informer: beyond efficient transformer for long sequence time-series forecasting," in *AAAI*, pp. 11106-11115, 2021.
[24] J. Luo, M. Ye, C. Xiao, and F. Ma, "Hitanet: hierarchical time-aware attention networks for risk prediction on electronic health records," in *SIGKDD*, pp. 647-656, 2020.
[25] H. Ren, J. Wang, W. X. Zhao, and N. Wu, "RAPT: pre-training of time-aware transformer for learning robust healthcare representation," in *SIGKDD*, pp. 3503-3511, 2021.
[26] G. Zerveas, S. Jayaraman, D. Patel, A. Bhamidipaty, and C. Eickhoff, "A transformer-based framework for multivariate time series representation learning," in *SIGKDD*, pp. 2114-2124, 2021.
[27] A. Johnson, L. Bulgarelli, T. Pollard, S. Horng, L. A. Celi, and R. Mark, "Mimic-iv (version 0.4)," in *PhysioNet*, 2020.
[28] T. J. Pollard, A. E. Johnson, J. D. Raffa, L. A. Celi, R. G. Mark, and O. Badawi, "The eICU collaborative research database, a freely available multi-center database for critical care research," in *Scientific Data*, vol. 5, no. 1, 2018.
[29] A. Johnson, J. Aboab, J. D. Raffa, T. J. Pollard, R. O. Deliberato, and L. A. Celi et al., "A comparative analysis of sepsis identification methods in an electronic database," in *Critical Care Medicine*, vol. 46, no. 4, 2018.
[30] D. J. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series," in *KDD workshop*, vol. 10, no. 16, pp. 359-370, 1994.
[31] J. Y. Franceschi, A. Dieuleveut, and M. Jaggi, "Unsupervised scalable representation learning for multivariate time series," in *NeurIPS*, pp. 4652-4663, 2019.
[32] L. Van der Maaten, G. Hinton, "Visualizing data using t-SNE," in *Journal of Machine Learning Research*, vol. 9, no. 11, pp. 2579-2605, 2008.