Sepsis Detection Using Missingness Information

Clémentine Aguet[1,2], Jérôme Van Zaen[1], Mathieu Lemay[1]

[1] Swiss Center for Electronics and Microtechnology (CSEM), Neuchâtel, Switzerland
[2] Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland

Abstract

Sepsis is one of the leading causes of death in hospital. An early detection is crucial to reduce its consequences and mortality. The challenge of Computing in Cardiology 2019 is addressing this issue by providing about 40,000 records from intensive care unit patients. As clinical measurements are collected at irregular frequencies, this dataset is missing many observations. Simply discarding missing values is counterproductive. Indeed, it has been observed that missing data patterns hold relevant information regarding the patient health state. To take advantage of this information, we propose a sepsis detection model incorporating representations of missingness information. This model is a recurrent neural network network composed of two gated recurrent unit (GRU) layers to capture long-term dependencies and a sigmoid layer to output a probability of sepsis. First, the model is trained by simply imputing missing values in the dataset. Then, the dataset is extended with the pattern of missing values. Finally, the model with a modified GRU cell taking into account missing data is evaluated. Our best model achieves an utility score of 0.00 on the final test set.

1. Introduction

Sepsis is a critical multifaceted complication of an infection. It is a life-threatening condition induced by an overactive response of the body [1]. Instead of fighting invaders, the immune system starts attacking itself. If not detected early enough or without proper treatment, it may trigger damages and dysfunctions in tissues and organs. Even though the worldwide impact of sepsis remains complicated to establish, it has been stated by the World Health Organization [2] to affect every year more the 30 million people with 6 million potential deaths.

In comparison to other diseases, no gold standard test exists for sepsis diagnosis. It is typically identified from a set of signs, symptoms and test results. Some clinical tests can help to detect biomarkers of sepsis [3,4]. However, diagnosing sepsis early remains challenging which, in turn, delays the selection of the optimal treatment. The challenge of Computing in Cardiology 2019 [5] is addressing the issue of early sepsis detection by providing a dataset including vital signs, laboratory test results, demographics collected from intensive care unit patients in different hospitals.

In healthcare, medical records typically include observations measured irregularly over time, across variables and patients. Assuming that the interval between clinical measurements is linked to the rate at which vital parameters vary, the pattern of observations might carry rich information regarding a person's health. This concept is referred as *informative missingness*, indicating a correlation between target labels and missing rate or pattern. A relation verified recently in [6].

Handling missing values in a proper way is critical for the task performance and induces theoretical and computational challenges. The majority of machine learning models are not designed to take them into account. Different methodologies were presented to tackle this issue [7]. The simplest approach is probably to exclude missing data, which consequently reduces the number of training samples available. Another possibility is data imputation, which consists of replacing gaps with substitute values. It can be substituting the missing observation by the mean of the variable computed across the training samples. By taking into account the temporal characteristic of time-series and presuming that a medical record is almost identical as its previous measurement, missing values can be replaced using forward imputation. However, *informative missingness* is not exploited with such approaches. Lipton et al. [8] proposed to add a binary mask to indicate missing values and the delays since the previous measurements to the inputs. Che et al. [6] suggested another method to deal with missing data. They incorporate a decay mechanism to a *gated recurrent unit* (GRU) and use masking and time intervals as missing pattern representations.

The main goal of this work is to investigate approaches dealing with missing values for *recurrent neural networks* (RNN) and explore the potential of missingness information for the task of sepsis detection.

2. Methods

2.1. Dataset

The training data for the challenge includes 40,336 records from ICU patients of two different hospitals. Data from another hospital system remains censored and are used for grading. The data of the n-th patient, with $n \in \{1, ..., N\}$, is represented as multivariate time-series $X_n \in \mathbb{R}^{T \times D}$ with D variables of length T. It combines 40 variables including demographics, vital signs and laboratory measurements. The vital signals and laboratory measurements are sampled every hour. The value x_t^d denotes to the t-th observation of the d-th variable. A binary target $y_t \in \{0, 1\}$ is associated with each time interval. It indicates the onset of sepsis according to the Sepsis-3 definition [1], with 0 for non-sepsis and 1 for sepsis. It is important to mention that the sepsis label is shifted by six hours. The task of the challenge is to predict sepsis six hours ahead.

Several aspects of this dataset are challenging. In particular, the irregularity of clinical measurements leads to many missing values in the dataset. Some variables are entirely missing for some patients, while others have different sampling frequencies across patients and time. Treating them appropriately is likely a key element for accurate sepsis prediction. Another issue is the imbalanced classes, with only 7.3% sepsis patients over the whole training dataset.

2.2. Data Preprocessing

Features are extracted based on their relevance for sepsis detection task and their missing rate. Features used in the Sepsis-3 definition [1] are included. Then, a feature is excluded if either its average missing rate over the training set is larger than 93% or it has no measurement for 90% of the patients. In total, 18 features were selected, including vital signs, laboratory values, and demographics. They are listed in Table 1 and described in [5].

Table 1. List of selected features with references.

Vital signs	Laboratory	Demographics
HR - 90 bpm	HCO3 - 26 mmol/L	Age
O2Sat - 97.5 %	FiO2 - 0.21 mmol/L	Gender
Temp - 37 °C	pH - 7.4	ICULOS
SBP - 120 mmHg	Creatinine - 0.9 mg/dL	
MAP - 94.7 mmHg	Glucose - 90 mg/dL	
DBP - 80 mmHg	Hgb - 15 g/dL	
Resp - 16 breaths per min	WBC - 7.7 $10^3/\mu L$	
	Platelets - 275 $10^3/\mu L$	

The following pre-processing steps are applied before feeding the inputs to the model. For vital signs and laboratory values, normalization is performed using a reference value, which are defined by experts [9] for each feature, see Table 1. For demographic variables, the mean computed over the training set is used for normalization.

Representations of missingness are associated with each multivariate time series X_n. A binary mask is used to indicate which features are observed or missing at each time step. In addition, a array of time interval reveals the duration since the last observation for each features. The binary mask of the n-th patient, $M_n \in \{0, 1\}^{T \times D}$ is composed of elements m_t^d defined as follows

$$m_t^d = \begin{cases} 1 & \text{if } x_t^d \text{ is observed,} \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

Each entry δ_t^d of the corresponding time intervals $\Delta_n \in \mathbb{R}^{T \times D}$ is computed as

$$\delta_t^d = \begin{cases} s_t - s_{t-1} + \delta_{t-1}^d & t > 1,\ m_{t-1}^d = 0 \\ s_t - s_{t-1} & t > 1,\ m_{t-1}^d = 1 \\ 0 & t = 1 \end{cases} \quad (2)$$

where s_t represents the time-stamp of the t-th observation.

In a first approach, missing values are imputated following two methods. Based on the assumption that a variable without any measurement is probably considered by the clinical staff to be in the physiological range, it is substituted by a reference value x_{ref}^d.

$$x^d = x_{ref}^d \quad \text{if } m_t^d = 0\, \forall t$$

Otherwise, it is the temporal structure which is exploited. A clinical measurement is assumed to be almost identical to the previous one. Missing values are imputated in a fill forward manner using the last observation $x_{t'}^d$.

$$x_t^d = x_{t'}^d \quad \text{if } m_t^d = 0$$

In a second approach, as the idea is to consider the missing pattern, gaps are filled with zeros for computational purposes.

The 40,336 records are partitioned into training, validation, and test sets. The subsets are stratified with the sepsis label, in order to have the same proportion of sepsis and non-sepsis cases in each set. The train set (70% of the records) is used to fit the model. The validation set (15% of the records) to tune the hyper-parameters. The remaining 15% of the records are kept to finally evaluate the model on unseen data.

During the learning and optimization process, instead of loading the whole training set at each iteration, a batch of samples is given to the model. As the sequences do not share the same length, zero-padding is required to equalize the sequence length in each batch.

2.3. Sepsis Detection Model

A model based on a RNN was selected as this class of networks have achieved state-of-the-art performance on numerous time series processing tasks. Such neural networks were designed to capture temporal dependencies and to handle sequences of different lengths. Its outputs are recursively calculated from given inputs x_t and previously computed states h_t. GRU [10], a type of RNNs, was designed to adapt and capture dependencies at different time scales. The potential of such model is investigated here for early sepsis detection.

The baseline model implemented for the challenge takes as inputs the imputated data $[X_n]$. It is a RNN composed of two GRU layers with 100 hidden units each and a dropout rate of 0.3, followed by a fully connected layer with sigmoid activation function as prediction layer. The model is trained over 30 epochs by minimizing the cross-entropy with the Adam optimizer [11] having a learning rate of 0.002. As the number of records with and without sepsis are not balanced, the updates for each class are weighted. The weights are computed by dividing the number of samples of the largest class (here non-sepsis) by the number of samples of the given class. In addition, the following hyper-parameters are tuned with Bayesian optimization: number of GRU layers, number of hidden units, and learning rate.

The second approach is based on [8] and adds a binary mask indicating missing features (1). Data and mask are concatenated into a single vector $[X_n; M_n]$ that is fed to the network. A model similar to the previous one with equivalent parameters is trained using this augmented features space. This model is referred to as *GRU-mask* in the rest of the paper.

Clinical data are typically characterized by the fact that inputs have less impact and tend towards a given value when last observations happen a long time ago. Such properties can be captured using a decay mechanism and are introduced in a modified version of a GRU cell, similar to the one proposed in [6]. The structure of the GRU decay (GRU-D) cell is illustrated in Figure 1.

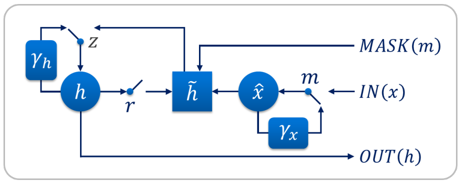

Figure 1. GRU-D cell

There are two decay rates, which are computed following the equation below. The idea is to learn them during the training and not to have them previously defined.

$$\gamma_t = \exp\left\{-\max\left(0, W_\gamma \delta_t + b_\gamma\right)\right\}$$

First, inputs are decayed over time toward the empirical mean $\tilde{x}^d$. Which is computed over the whole training set and applied to the training and test sets. The input decay rates are ensured to be independent for each variable by forcing γ_{x_t} to be diagonal.

$$\hat{x}_t^d = m_t^d x_t^d + \left(1 - m_t^d\right)\left(\gamma_{x_t}^d x_{t'}^d + \left(1 - \gamma_{x_t}^d\right)\tilde{x}^d\right)$$

To gain further information from the missingness pattern, a decay term is also applied to the extracted features, known as hidden states.

$$\hat{h}_{t-1} = \gamma_{h_t} \odot h_{t-1}$$

The GRU-D update functions are defined as follows.

$$r_t = \sigma(W_r \hat{x}_t + U_r \hat{h}_{t-1} + V_r m_t + b_r)$$

$$z_t = \sigma(W_z \hat{x}_t + U_z \hat{h}_{t-1} + V_z m_t + b_z)$$

$$\tilde{h}_t = tanh(W \hat{x}_t + U(r_t \odot \hat{h}_{t-1}) + V m_t + b)$$

$$h_t = (1 - z_t) \odot \hat{h}_{t-1} + z_t \odot \tilde{h}_t$$

A final model based on this modified GRU cell is composed of two GRU-D layers with 100 hidden units followed by a fully connected layer with sigmoid activation function as prediction layer. For comparison purposes, it is trained in a similar manner as the models described previously. This model takes as inputs the data, the mask and the time interval $[X_n; M_n; \Delta_n]$.

2.4. Model Evaluation

All models are evaluated on the same training, validation, and test partitions. The metrics reported are the area under the receiver operating characteristic curve (AUROC), the area under the precision-recall curve (AUPRC), the accuracy, the F-measure, and the utility score. This last metric was created for the challenge [5]. The utility score rewards early predictions while penalizing erroneous, late or missed predictions and takes values in the range $[-2, 1]$. Final grading is performed by the challenge organizers using the utility score and a private dataset collected in a third hospital.

3. Results

The three model configurations are trained and evaluated on the challenge data. Their performance metrics measured on the training, validation, and test set are reported in Table 2. GRU and GRU-mask models obtain similar scores on the training and validation sets. However, the scores computed on the test set indicate that the mask helped to reduce overfitting. This effect is clearly marked for the utility score which decreases sharply for the GRU model. The GRU-D model achieved higher AUROC and utility scores than the two previous models. While, its accuracy was lower. It seems that GRU-D model is the most resilient to overfitting as the scores obtained on the three sets are very similar. However, it did not generalize well on data from an other hospital as it achieved an utility score of 0.00 on the set used for ranking challenge entries.

Table 2. Performance on training, validation and test sets.

Metrics	GRU	GRU-mask	GRU-D
Training			
AUROC	0.70	0.70	0.78
AUPRC	0.08	0.08	0.08
Accuracy	0.89	0.89	0.78
F-measure	0.12	0.12	0.10
Utility	0.26	0.27	0.30
Validation			
AUROC	0.70	0.67	0.76
AUPRC	0.09	0.09	0.08
Accuracy	0.83	0.90	0.76
F-measure	0.09	0.14	0.08
Utility	0.22	0.25	0.29
Test			
AUROC	0.67	0.68	0.76
AUPRC	0.09	0.10	0.07
Accuracy	0.83	0.93	0.73
F-measure	0.08	0.16	0.09
Utility	0.16	0.26	**0.29**

4. Discussion

This paper presents an approach for the task of early sepsis detection in the context of the challenge of Computing in Cardiology 2019. The potential of GRU networks and missingness information is investigated for this problem. From the given dataset, 18 features from vital signs, laboratory values, and demographics are selected for this task. Representations of missing values can be associated to each record in the form of a binary mask and time intervals between measurements. They are here incorporated to the model either as additional features (GRU-mask) or by using decay mechanisms (GRU-D). The performance metrics computed on the training, validation, and test sets indicate that including missingness information helps to reduce overfitting. In particular, the GRU-D model obtained a utility score of 0.29 on the test set, almost identical to the values obtained on the training and validation sets. However, it did not generalize well on the final dataset.

Since this dataset is from another hospital, it is likely that the performance is a bit lower than the one obtained on the test set from available data. The gap between test and final scores indicates the presence of overfitting and poor generalization. GRU-D model achieves good performance as long as the patient comes from a hospital from which data where used during the training. Taken together, these results suggest that the procedures for diagnosing sepsis might be different across institutions. No specific guideline exists for recordings and decisions are often took based on clinicians intuitions.

Even though the outcomes show some overfitting on the final dataset, the approach of informative missingness remains promising and further investigations should be carried on. Training such model involves many hyperparameters which affect the global performance. In order to enhance the network, extensively tuning the hyperparameters should be considered. Changing the network structure or adding more features might also help to improve the performance and reduce overfitting.

References

[1] Singer M, et al. The third international consensus definitions for sepsis and septic shock (sepsis-3). JAMA 2016; 315(8):801–810.

[2] Word Health Organization. Sepsis. `https://www.who.int/news-room/fact-sheets/detail/sepsis`, 2018.

[3] Fan SL, S. Miller N, Lee J, Remick D. Diagnosing sepsis – the role of laboratory medicine. Clinica Chimica Acta 2016;460:203–210.

[4] Faix J. Biomarkers of sepsis. Critical reviews in clinical laboratory sciences 2013;50:23–36.

[5] Reyna M, Josef C, Jeter R, Shashikumar S, Westover M, Nemati S, Clifford G, Sharma A. Early prediction of sepsis from clinical data: the physionet/computing in cardiology challenge 2019. Critical Care Medicine in press;.

[6] Che Z, Purushotham S, Cho K, Sontag D, Liu Y. Recurrent neural networks for multivariate time series with missing values. Scientific Reports 2016;8(1):6085.

[7] L. Schafer J, Graham J. Missing data: Our view of the state of the art. Psychological Methods 2002;7(2):147–177.

[8] Lipton Z, Kale D, Wetzel R. Modeling missing data in clinical time series with rnn. arXiv preprint arXiv160604130 2016;.

[9] American Board of Internal Medicine. Abim laboratory test reference ranges. `https://www.abim.org/~/media/ABIM%20Public/Files/pdf/exam/laboratory-reference-ranges.pdf`, 2019.

[10] Cho K, van Merriënboer B, Gulcehre C, Bougares F, Schwenk H, Bengio Y. Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv preprint arXiv14061078 2014;.

[11] Kingma DP, Ba J. Adam: A method for stochastic optimization. arXiv e prints 2014;arXiv:1412.6980.

Address for correspondence:

Clémentine Aguet
CSEM SA
Rue Jaquet-Droz 1, 2002 Neuchâtel, Switzerland
clementine.aguet@csem.ch