Strategies to improve the performance of neural networks for early detection of sepsis

ByeongTak Lee, KyungJae Cho, Oyeon Kwon and Yeha Lee

VUNO, Seoul, South Korea

Abstract

Objectives. *Early detection of sepsis is a clinically important, yet remains challenging. As machine learning develops, there have been many approaches for prediction of sepsis using neural network-based models. In this work, We propose novel preprocessing and training strategies, which can boost the performance of the model.*

Methods. *Our approach consist of two-component: feature engineering and training strategy. In feature engineering, we employed a novel input imputation method that combines input decay, masking, and duration of missing and input transformation which is the value difference between adjacent time step. As for training strategy, manipulation of the normal distribution, reconstruction loss, and population-based training are utilized.*

Results. *We applied our method on three-block transformer. Our approach achieved an AUROC/AUPRC of 0.819/0.144, Physionet Challenge score of 0.376 on the held-out test. Compared to the baseline, we improved performance AUROC of 0.034, AUPRC of 0.024, and score of 0.079. On the hidden test, we obtained the score of 0.387.*

Conclusion. *We have developed a novel approach that can improve the performance of neural networks in the early detection of sepsis.*

1. Introduction

Sepsis is a life-threatening disease caused by an uncontrolled response to infection. In worldwide, an estimated 30 million people develop sepsis, and 20 percent of them die from it every year [1]. Missing golden time for appropriate treatment is considered as the main reason for mortality [2]. For this reason, early identification is critical for improving sepsis outcomes, yet remains challenging [3].

As machine learning technologies develop, there have been many studies applying machine learning to predict sepsis. Early studies concentrated on a statistical methods [4,5]. Lately, there have been efforts to apply a deep neural network to early detection of sepsis, and they outperform classical statistical methods [6, 7].

Whereas studies on building architectures continue, preprocessing, or training strategies for sepsis prediction has not been studied thoroughly. In this work, we suggest s novel preprocessing method, combining input imputation methods and data transformation method. In the meantime, we also adopted a novel training strategy, including normal data resampling and population-based training. Utilizing the methods described in the paper, we obtained meaningful improvement in performance with three-block transformer.

2. Methods

In this chapter, we describe the database we employed in developing the model, the structure of the neural network, methods to improve the neural network, and evaluation method. In the section of the structure of neural network, the baseline model and hyperparameters for training the model are introduced. Afterward, approaches we adopted to improve the performance are presented in following section.

2.1. Data

We developed the model using the database from Physionet Challenge 2019. The database is collected from two hospitals consisted of 20,336 and 20,000 patients, respectively, of which 1,790 and 1,142 patients underwent sepsis in each hospital. Forty variables, including vital, laboratory, and demographics, were used as predictor variables. Detailed information about the database can refer to [8].

2.2. Model

Transformer The architecture adopted in this work is based on a multi-layer bidirectional Transformer encoder described in [9]. In this work, we used three-layer instead of six-layer of original paper because there was no performance improvement as the number of layer increases in our experiment. Except for the number of layers, implementation is identical to the original.

Hyperparameters Hyperparameters adopted in the

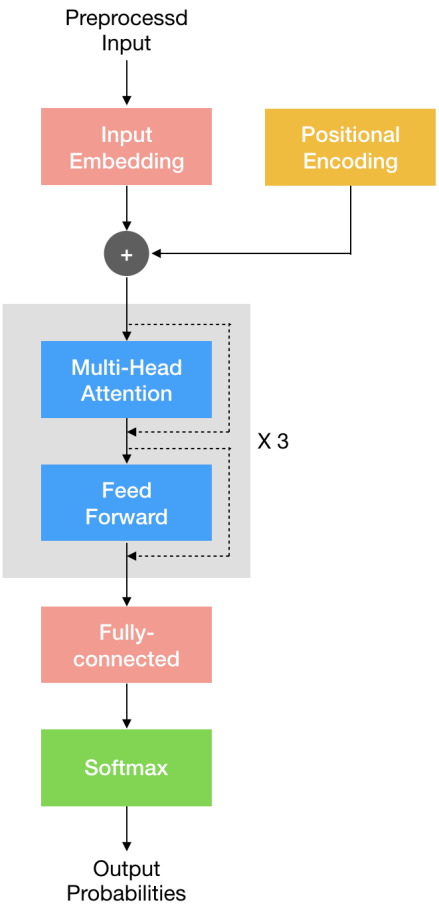

Figure 1. An overview of the architecture. Positional encoding and three-layer of transformer are utilized. The detail of architecture can refer to [9].

training the model are as follow. We used the learning rate of 1e-3 with Adam optimize. The batch size for training is 32. Dropout and weight decay are employed for generalization of the model. The dropout rate in transformer and classifier are 0.1 and 0.5, respectively. Also, L2 regularization with multiplication by 0.001 is applied.

2.3. Proposed methods

We improved the three-block transformer by applying the following approaches. At first, we explored various methods for input imputation and transformation methods and elaborated by combining them. Next, various techniques, including data sampling, reconstruction error, and population learning, are developed.

Original values	70	75	X	X	X	88	X	75
Imputed values	70	75	76	79	80	88	84	75
Difference transformation	0	-5	1	3	1	8	-4	-9
Masking	1	1	0	0	0	1	0	1
Duration of missing	1	1	2	3	4	1	2	1

Figure 2. Example of feature engineering. The first row and following four rows refer original values and the example of input imputation and transformation.

2.3.1. Preprocessing

Data scaling variables of Electro Medical Records(EMR) data have different units and range. For example, Heart rate ranges from 60 to100 in the normal case, while normal pH ranges from 7.38 to 7.42. These differences in the scales across input variables increase the difficulty in training the model [10].To solve the problem, we applied standardization and used cutoff values sigma 5.

Feature engineering EMR data unavoidably contains missing observation induced by medical events, abnormalities, and inconvenience. There are several methods to handle missing values of EMR data, including forward-imputation, mean-imputation, and utilization of masking [11, 12]. Forward-imputation assumes missing values as same as its last measurement. Masking is used to distinguish the true values from imputed values, which is often applied with a duration of missingness. Recently, [13] suggested a novel missing value imputation model that decays the missing value to zero as the difference between its last observation and current time increases. We applied combinations of input imputation methods, including method [13], masking, and duration of missingness in this work. As variables of EMR reflects the status of patients, they tend to be non-stationary, which is more challenging to learn the attributes of the sequence [14]. Thus, we calculate difference between adjacent time step of each variable. It removes the temporal dependence such as time-series trend, making the sequence stationary [15].

2.3.2. Training strategies

Data resampling The model trained on the skewed distribution of class tends to have difficulty in learning the properties of a minority class and is biased to the majority class [16]. In our dataset, the number of the sepsis label is only 1 of 30 of the normal label, which probably leads to the difficulty in training described above. To handle this

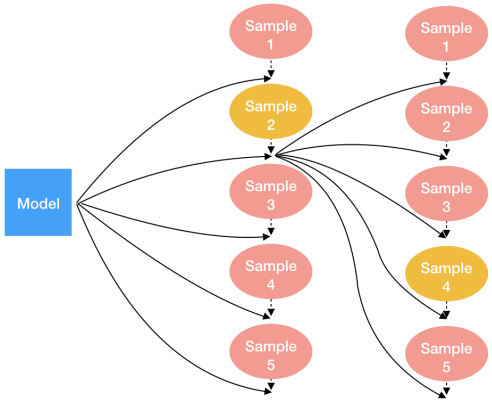

Figure 3. The example of population-based learning. The model is trained on different five set of samples and the model with highest performance is selected.

problem, we undersampled minority class data so that balanced the class ratio in the training sample. Furthermore, we manipulate the distribution of normal data point. The ratio between normal data point from sepsis patient and normal data point from a non-sepsis patient is 1:8. We undersampled the normal data points from patients who underwent sepsis and make the ratio of those data as 1:4 in training. It allows the model to be trained frequently on hard samples, inducing the model to be more robust.

Reconstruction loss The primary loss function employed in training is the cross-entropy function. Additionally, we adopted reconstruction loss to prevent the model from overfitting on a training dataset. The reconstruction error is calculated as the L2 distance between imputed input with different transformed value and linearly transformed output of the transformer.

Population-based learning We found that the performance of the model decreases whenever a specific portion of the dataset flow into the training process. It means that some part of data deteriorate training. We parallelly trained the model on five subsets of randomly selected samples from the bootstrapped data pool, and choose the model with the highest performance among them to handle the problem. At every epoch, the parallel training begins from the best model of the previous epoch.

2.4. Evaluation

In model evaluation, we hold out a randomly selected 20 percent of overall data as an internal test. The rest dataset of 80 percent is divided into training and development dataset. The ratio of hospital A and B are equal in each set. As for performance metrics, we used Area Under Receiver Operator Curve (AUROC), Area Under Precision-Recall Curve (AUPRC), and score function provided by Physionet Challenge 2019. During model training, we evaluated the performance using a sum of AUROC and AUPRC. Once training is done, we compute the score function, which is adjusted by threshold values. The threshold value with the highest score function is a cutoff value for the model. Afterward, the model is tested on an internal test that was held out from the training process. We choose the model based on the internal test score, and then tested on the hidden external test.

3. Results

We fixed the model as a three-layer transformer and trained on various combination of preprocessing and training strategies. Table 1 represents the matrix of a combination of them. Each row indicates the preprocessing methods and the columns indicate training strategies.

Blue-colored sections mean the three highest methods based on score, and the red-colored sections mean the three lowest methods. There is a trend that performance increases as training strategies and preprocessing methods are applied. The proposed model achieved the highest performance with the AUROC/AUPRC of 0.819/0.144 and score of 0.376. On the other hand, the performance of the baseline is one of the lowest, which is AUROC/AUPRC of 0.785/0.120 and score of 0.297. The performance gain using the suggested method was AUROC of 0.034, AUPRC of 0.024, and score of 0.079. Finally, we tested the model on a hidden external test and obtained a score of 0.387.

4. Conclusions and Discussion

In this work, we present various input processing and training strategy for sepsis prediction from clinical data. As an input preprocessing, [13] based imputation, difference transformation, masking, and duration of the missing are employed. Meanwhile, the sampling method from the dataset, reconstruction loss, and population training are implemented as a training strategy. Using the proposed method, we gained meaningful performance improvement with the 3-block of the transformer. In the internal test dataset, our model achieved the AUROC of 0.819, AUPRC of 0.144, and Physionet Challenge score of 0.376, and it obtained Physionet Challenge score of 0.387 in the external dataset.

The suggested methods are tested with hyperparameters fixed. We expect that hyperparameter optimization such as learning rate, weight decay rate, optimization method, and reconstruction loss ratio, could further enhance the performance.

When comparing the model performance in the internal test set, we observed that the scores of several models

		Training strategy				
	AUROC/AUPRC SCORE	Base	Resampling (between normal point of sepsis patient and the other patients)	Reconstruction loss	Population training	Proposed
Preprocessing	Base (Forward-imputation)	0.785/0.120 0.297	0.785/0.124 0.293	0.789/0.127 0.294	0.788/0.123 0.308	0.789/0.120 0.309
	Forward-imputation, masking, duration of missing	0.808/0.138 0.347	0.811/0.138 0.341	0.814/0.137 0.366	0.806/0.141 0.356	0.816/0.140 0.368
	Proposed	0.810/0.138 0.359	0.813/0.140 0.365	0.818/0.143 0.366	0.811/0.142 0.360	0.819/0.144 0.376

Table 1. The result of proposed methods. The red-colored section indicates three lowest-score and the blue-colored section indicated three highest-score. It is found that proposed methods enhance the performance of transformer significanlty.

are very sensitive to a threshold value. High sensitivity of score around the cutoff value indicates that there are many data points close to the decision boundary. A slight deviation in the overall distribution of the data can significantly degrade the performance in this case. Further investigation about the association between distribution of data point and the generalization ability of the model is required, which we intend to study in the future.

References

[1] Singer M, Deutschman CS, Seymour CW, Shankar-Hari M, Annane D, Bauer M, Bellomo R, Bernard GR, Chiche JD, Coopersmith CM, et al. The third international consensus definitions for sepsis and septic shock (sepsis-3). Jama 2016;315(8):801–810.

[2] Seymour CW, Gesten F, Prescott HC, Friedrich ME, Iwashyna TJ, Phillips GS, Lemeshow S, Osborn T, Terry KM, Levy MM. Time to treatment and mortality during mandated emergency care for sepsis. New England Journal of Medicine 2017;376(23):2235–2244.

[3] Paoli CJ, Reynolds MA, Sinha M, Gitlin M, Crouser E. Epidemiology and costs of sepsis in the united states—an analysis based on timing of diagnosis and severity level. Critical care medicine 2018;46(12):1889.

[4] Calvert JS, Price DA, Chettipally UK, Barton CW, Feldman MD, Hoffman JL, Jay M, Das R. A computational approach to early sepsis detection. Computers in biology and medicine 2016;74:69–73.

[5] Desautels T, Calvert J, Hoffman J, Jay M, Kerem Y, Shieh L, Shimabukuro D, Chettipally U, Feldman MD, Barton C, et al. Prediction of sepsis in the intensive care unit with minimal electronic health record data: a machine learning approach. JMIR medical informatics 2016;4(3):e28.

[6] Kam HJ, Kim HY. Learning representations for the early detection of sepsis with deep neural networks. Computers in biology and medicine 2017;89:248–255.

[7] Moor M, Horn M, Rieck B, Roqueiro D, Borgwardt KM. Temporal convolutional networks and dynamic time warping can drastically improve the early prediction of sepsis. CoRR 2019;abs/1902.01659. URL http://arxiv.org/abs/1902.01659.

[8] Reyna MA, Josef C, Jeter R, Shashikumar SP, M. Brandon Westover MB, Nemati S, Clifford GD, Sharma A. Early prediction of sepsis from clinical data: the PhysioNet/Computing in Cardiology Challenge 2019. Critical Care Medicine 2019;In press.

[9] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. Attention is all you need. In Advances in neural information processing systems. 2017; 5998–6008.

[10] Bishop CM. Neural Networks for Pattern Recognition. New York, NY, USA: Oxford University Press, Inc., 1995. ISBN 0198538642.

[11] Lipton ZC, Kale D, Wetzel R. Directly modeling missing data in sequences with rnns: Improved classification of clinical time series. In Machine Learning for Healthcare Conference. 2016; 253–270.

[12] Choi E, Bahadori MT, Schuetz A, Stewart WF, Sun J. Doctor ai: Predicting clinical events via recurrent neural networks. In Machine Learning for Healthcare Conference. 2016; 301–318.

[13] Che Z, Purushotham S, Cho K, Sontag D, Liu Y. Recurrent neural networks for multivariate time series with missing values. Scientific reports 2018;8(1):6085.

[14] Jung K, Shah NH. Implications of non-stationarity on predictive modeling using ehrs. Journal of biomedical informatics 2015;58:168–174.

[15] Hyndman RJ, Athanasopoulos G. Forecasting: principles and practice, 2013. URL httpswww otexts orgfpp accessed 2018 02 15WebCite Cache ID 6xFJlXCQI 2017;.

[16] Johnson JM, Khoshgoftaar TM. Survey on deep learning with class imbalance. Journal of Big Data 2019;6(1):27.