Team 11: Sepsis Prediction
CSE 6250 Project Report

Github: https://github.gatech.edu/hlee873/CSE6250_Team11

Presentation: https://youtu.be/90_8Q0_a064

Members: Hao Lee, Michael Hur, Qasim Nazir, David Wu

GTIDs: hlee873, hurm, qnazir3, dwu89

Abstract

We focus on predicting sepsis, a medical condition where the immune system damages the body as a result of fighting infection. We introduce and replicate a Long Short-Term Memory (LSTM) neural network model that uses patient features from the Medical Information Mart for Intensive Care (MIMIC)-III dataset for early identification and prediction of sepsis, as defined by Sepsis-3. Ultimately, this report highlights the approach and results for sepsis prediction strategy.

Introduction

Sepsis is one of the leading causes of death in the United States and includes the highest mortality rates among patients who develop septic shock [5]. This medical condition is caused by a dysregulated immune response to an infection [3]. Consequently, if left unchecked, sepsis can progress to septic shock, where the body experiences organ failure and lowered blood pressure with a high mortality rate. To combat sepsis, while there are general-purpose illness severity scoring systems and mortality prediction, there lacks a highly sensitive and specific prediction system unique to acute sepsis [5]. Hence, the benefit of reliable sepsis identification and prediction will optimize early treatment for effective results, thereby saving lives. In this study, we used the MIMIC-III dataset and created a deep learning neural network to detect patients with Sepsis-3.

Literature Review

Various machine learning algorithms have been experimented with for early prediction of sepsis and septic shock. Henry et al. [5], in 2015, proposed a real-time early warning score (TREWScore) to predict the onset of septic shock, using supervised machine learning with MIMIC-III data. TREWScore identified patients before the onset of septic shock hours in advance and with AUROC of 0.85. With 0.67 specificity, TREWScore achieved 0.85 sensitivity and detected septic shock at a median of 28.2 hours earlier. Another predictive classification model, Insight, was developed by Calvert et al. [2] in 2016 for early sepsis onset prediction based on the Sepsis-3 definition with MIMIC-III data. In a test dataset with 11.3% sepsis prevalence, Insight produced AUROC of 0.88 1-4 hours before sepsis onset and achieved 0.595 in precision-recall curves. The latest machine learning model to predict sepsis was developed by Liu et al. [10] in 2019. The approach developed many machine learning algorithms (GLM, XGBoost, and RNN) to model risk of septic shock, finding that the GLM yields the most interpretable model and results, while the RNN has the highest AUC score at 0.93.

Other approaches for sepsis prediction utilize deep neural networks. In 2017, Harutyunyan [4] et al. created a multitask LSTM model to predict many clinical problems including the detection of sepsis, using MIMIC-III data. Additionally, Fagerstorm et al. [3] proposed a Long Short-Term Memory neural network (LiSep) in 2019 to predict the onset of Sepsis-3 by using MIMIC-III data. LiSep LSTM replicated the conditions outlined in the TREWScore study, using the features and targets to compare LiSep LSTM

to TREWScore. LiSep LSTM had an AUROC of 0.83 and an HBO median of 48, both scores higher than those of the TREWScore. Lastly, Kam et al. [9] proposed a feed-forward neural network for early detection of sepsis, achieving an AUROC score of 0.92.

Approach / Metrics

Data

Our data is procured from the MIMIC-III database [6], where it has been de-identified for patient health information and includes dimensions specific to patient demographics, diagnosis, procedures, medication, vital signs, lab results, notes, and more. We also have utilized the derived tables.

ETL Process

Before proceeding with the experimental setup and designs, the data needed to be transformed through an ETL process that makes the data suitable for the subsequent machine learning models. Below is a visual representation of the ETL process:

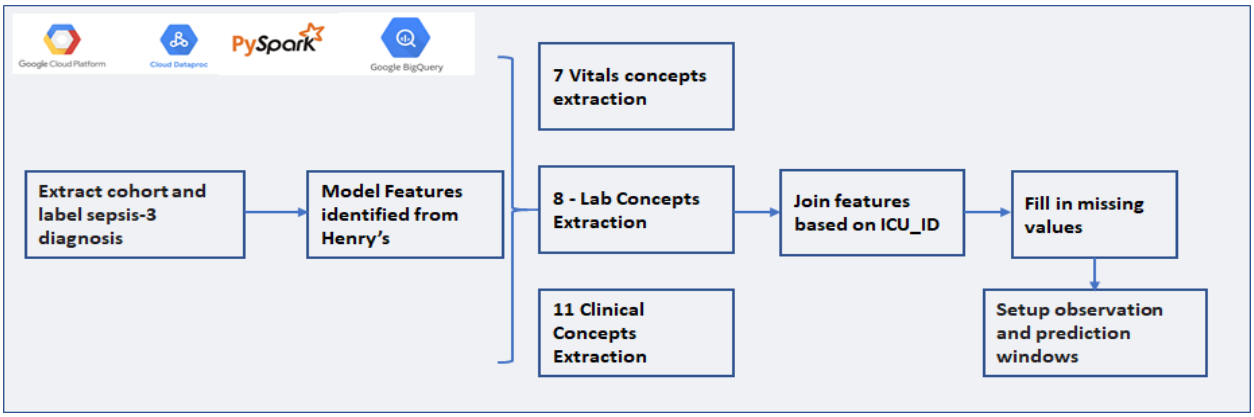

Figure 1: MIMIC-III ETL Process

(i) Extract Cohort labels: Our cohort was guided via [7], and our flowchart is below. One difference is the last step, where we took a 48 hour pre and post 24 hour window surrounding the first suspected infection. We included the ICU Stays where an overlap does not include at least 12 hours worth of data as we want enough data points for Sofa scoring. Further, we also removed edge cases where length of stay was less than 12 hours. Finally, we converted our cohort dataset into a time series data set over the hours of a patient's ICU stay.

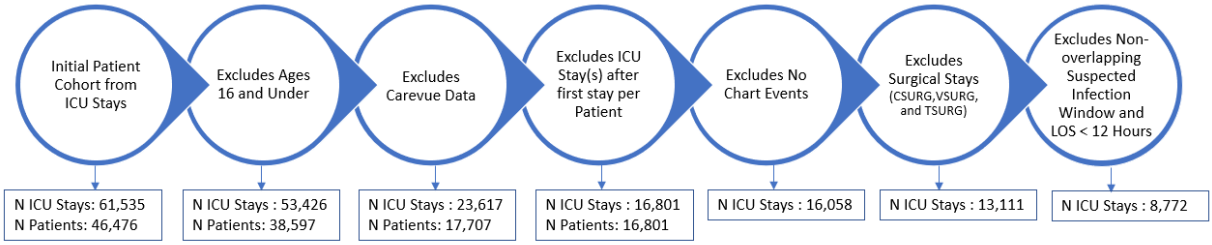

Figure 2: Cohort Label Extraction

(ii) Extracting Sepsis-3 Diagnosis Labels: The target label is whether a patient developed Sepsis-3. We define Sepsis-3 as the "Gold Standard" [2], based on a change in SOFA score greater than or equal to 2 within a 48 hour pre and 24 post hour window surrounding the first suspected infection time. This methodology differed from [7] as we looked beyond the first admission day. Our time series SOFA score is from a derived MIMIC-III table, named pivoted_sofa with a "sofa_24hours" score. This score is the max SOFA score over the last 24 hours by an hourly time stamp. With this methodology, we are able to determine whether an ICU stay has not only a Sepsis-3 diagnosis, but also the earliest time of onset.

(iii) Model features from Henry: The features for the RNN model are based on the predictors in Henry et al's study [5]. With a total of 26 features in the study, they are split into 7 vitals, 8 lab, and 11 clinical concepts. The concepts are generated using scripts published in the MIMIC-III Github. They are run through MIMIC-III BigQuery and are transformed using Pyspark on a Google Cloud Dataproc instance. Please refer to Appendix A for the Physionet table used to generate the concepts.

(iv) Join Features/Fill in Missing values/GroupBy: The vitals and lab concepts are averaged to an hourly level joined to the cohort by ICUSTAY_ID and Hour. Clinical concepts are joined by ICUSTAY_ID and Hour as well. Missing values are first forward and backward filled based on prior/future reading. Remaining missing values are filled by the population average. The final data is formatted in a table where the concept measurements are on an hourly basis. The final column Sepsis3_diag_flg indicates whether the patient had an onset of Sepsis-3.

icustay_id	age	gender	ethnicity	hour	timestamp	sofa_24hours	Sepsis3_start_flg	Sepsis3_diag_flg	sepsis_onset_hr
237810	26.10685	M	PATIENT DECLINED TO ANSWER	0	2186-01-28T07:00:00.000Z	0	0	1	16
237810	26.10685	M	PATIENT DECLINED TO ANSWER	1	2186-01-28T08:00:00.000Z	0	0	1	16
237810	26.10685	M	PATIENT DECLINED TO ANSWER	2	2186-01-28T09:00:00.000Z	0	0	1	16

Figure 3: Features and Labels Dataset

(v) Setting up Prediction and Observation Windows: We chose to vary both prediction and observation windows to better understand the range of model performance. Our set of prediction windows included (3,6,12) hours prior to the index date. We also tested observation windows (unlimited, 7, 12) hours. As for index date, for our cases, index date was set to the onset hour for Sepsis-3 diagnosis. For control, we chose to experiment with the last hour of the stay, and also used the average case Sepsis-3 diagnosis onset hour which was approximately 1/4th of the ICU Stay. After implementing the window, we also balanced the case and cohort population via random sampling without replacement from the control to downsample to match our case size.

Model Implementation for RNN Model Development

We used PyTorch framework to replicate the LSTM architecture [3] and trained neural network models with the newly processed patient data. Moreover, we used Google Colab (GPU) for training and evaluating models, and visualizing results. The raw features were copied to Google Drive and then loaded into the Colab notebook.

(i) Pre-processing/Custom Dataset /Zero-Padding & Dataloader: The pre-processing steps such as dropping redundant features, one-hot encoding categorical features and normalizing numerical features are applied to raw features.

A custom PyTorch Dataset is created by passing features dataframe (grouped by icustay_id) and overriding *__len__* and *__getitem__* methods. The *__getitem__* method retrieves features dataframe for a given index (icustay_id), applies pre-processing steps and returns *'sequence, label, id'*.

In this problem, input sequences are of varying lengths. Therefore, zero-padding is applied by passing a *'collate_fn'* to PyTorch Dataloader. The *'collate_fn'* also sorts sequences, labels & ids by sequence lengths. The data loader returns batches of *'(sequences, lengths), labels, ids'* tensors.

(ii) Building LSTM Model: LipSep [3] model was replicated along with a minor change as shown in Figure 4. It represents many-to-one RNN architecture for a binary classification problem. During the hyper-parameter tuning process, fully connected layers were added before and after the LSTM layer. The LSTM layer consisted of four layers with 100 LSTM units each and a dropout probability of 0.4. The default shape of input for recurrent layer modules in PyTorch is (seq_len, batch, input_size). We need to specify *'batch_first=true'* to make it compatible with shape of input tensor coming from dataloader. In the model *'forward'* function, torch's *'pack_padded_sequence'* and *'pad_packed_sequence'* utils are used. The final fully connected layer wrapped inside the 'sigmoid' function returns a probability of positive class (sepsis). A threshold value of 0.5 was used to predict class label during model evaluation.

```
MyLipSepFC(
  (fc1): Linear(in_features=31, out_features=1028, bias=True)
  (rnn): LSTM(1028, 100, num_layers=4, batch_first=True, dropout=0.2)
  (fcpost): Linear(in_features=100, out_features=32, bias=True)
  (out): Linear(in_features=32, out_features=1, bias=True)
)
```

Figure 4: Model Architecture

(iii) Model Training: The dataset(s) was divided into *train/validation/test* sets with ratios *0.75/0.1/0.15*. Following are the final training parameters that gave best performance.

Criterion	*Optimizer*	*Batch size*	*Cont. Learning rate*	*No of Epochs*
Binary Cross Entropy Loss	ADAM	3	0.0001	100

Table 1: Training Parameters

Apart from adobe parameters, we also tried following parameter combinations during the hyper-parameters tuning phase.

LSTM vs GRU	LSTM performed better
Adam vs SGD	Adam optimizer performed better
Learning Rate Scheduler	It did not help much.
BCELoss vs BCEWithLogitsLoss vs CrossEntropyLoss	Using a different loss function didn't help much. We decided to use BCELoss() as we wanted to get a probability of having sepsis as network output.

Table 2: Hyper-Parameter Tuning

Experimental Results

Trial #	Observation Window (Case)	Prediction Window (Case)	Index Date (Case)	Observation Window (Control)	Prediction Window (Control)	Index Date (Control)
1	Unlimited to Index Time	3	Sepsis Diagnosis Hour	Unlimited to Index Time	None	Last Hour of Stay
2	0 < Window Length <= 7 Hrs	3	Sepsis Diagnosis Hour	0 < Window Length <= 7 Hrs	None	Last Hour of Stay
3	0 < Window Length <= 12 Hrs	3	Sepsis Diagnosis Hour	0 < Window Length <= 12 Hrs	None	Last Hour of Stay
4	0 < Window Length <= 12 Hrs	3	Sepsis Diagnosis Hour	0 < Window Length <= 12 Hrs	None	Average Sepsis Diagnosis Hour
5	0 < Window Length <= 12 Hrs	6	Sepsis Diagnosis Hour	0 < Window Length <= 12 Hrs	None	Last Hour of Stay
6	0 < Window Length <= 12 Hrs	12	Sepsis Diagnosis Hour	0 < Window Length <= 12 Hrs	None	Last Hour of Stay

a) Trial Desriptions

Trial #	Accuracy	AOC	Sensitivity	Specificity	Precision
1	0.71	0.75	0.93	0.51	0.64
2	0.87	0.92	0.75	0.98	0.97
3	0.89	0.92	0.78	0.99	0.99
4	0.75	0.75	0.85	0.62	0.73
5	0.90	0.90	0.73	1.00	1.00
6	0.74	0.83	0.55	0.94	0.89

b) Predictive Metrics

Figure 5 :(a) Trial description (b) Predictive Validation Metrics

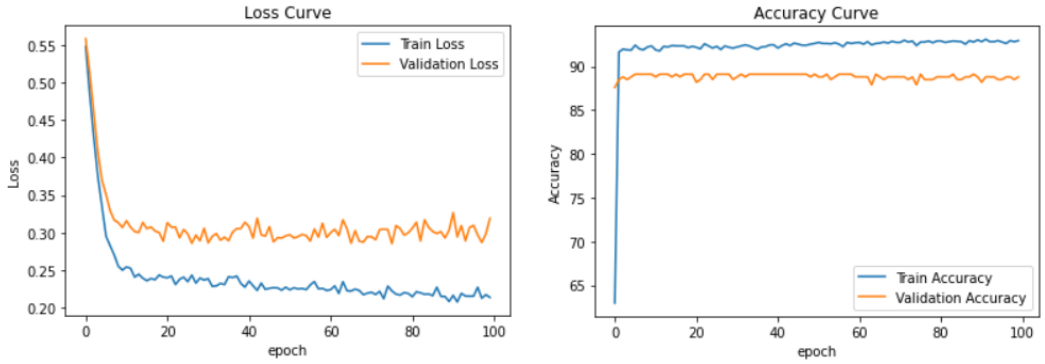

Figure 6 : Trial #3 Train/Validation Loss and Accuracy Curves

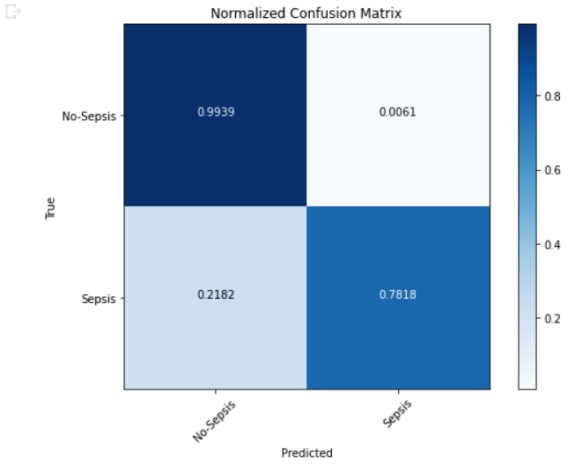

Figure 7 : Trial #3 Validation Confusion Matrix

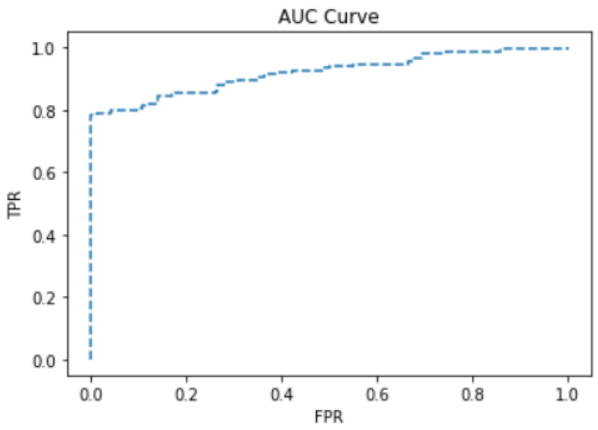

Figure 8 :Trial #3 Validation AUC Curve

Our models were trained for 100 epochs on Google Colab (GPU). The above figures show our validation results for Trial #3. Figure 5 shows that there exists a slight consistent overfitting behavior albeit the both train and validation are both relatively high compared to other trials. The loss and accuracy value for each epoch are computed as the average values for each ICU stay. Our resulting validation accuracy is

scored at 89%. The AUC is 0.92, and the sensitivity of the model is 78% and the specificity is higher at 99%.

Discussion

Challenges and Learnings

For the initial LSTM sequence model that we developed for our project draft submission, the biggest challenge that we faced was that our results showed low accuracy, an approximate AUC score of .5, and skewed negative Sepsis-3 labels. We had multiple failed attempts to boost our performance via hyper-parameter tuning, adding fully connected layers to the front and back of our LSTM architecture, adjusting our missing data method, and including additional clinical features. We eventually found that utilizing a balanced dataset, implementation of prediction and observation windows, and removal of edge cases led to significant improvement.

Our first discovery that led to improved results was the use of a balanced dataset as well as use of smaller batch sizes. We found that utilizing a balanced control and case population helped our LSTM sequence model to learn as initially our target balance was skewed towards our control, and thereby our model would predict skewed number of negative labels for Sepsis-3 relative to our cohort. Through downsampling our control population, our resultant balanced dataset allowed our model better discrimination. Further, we also found that utilizing a larger batch size did not uncover optimality that our smaller batch size was able to locate.

The use of prediction and observation windows or filters helped to curate relevant data. For example, we found that utilizing a larger observation window is not always optimal as a smaller window feeds more relevant data. Additionally, the choice of control index date also has an impact on our performance metrics as the control index date and associated observation windows reflect a patient's state during the ICU stay. For example, an index date set at the last hour with a 12 hour observation window may reflect the duration where the majority of patients are on the mend. Finally, we also restricted the length of icu stay to only include those greater than 12 hours. We also only included overlapping stays with our suspected infection window with a minimum of 12 hours. Removing these edge cases also helped performance.

Analysis of results and discussion of generalization

As noted in Figure 4, the use of different observation windows and index dates had a significant impact on performance. For observation windows, it is notable that sensitivity and specificity switches dominance when comparing trials one to trials two, three, five, six. In addition, it is also noteworthy to observe that in adjusting the control index date from trial three to trial four that our sensitivity and specificity also reversed in dominance. As for prediction windows, it appears that based on the prediction windows that were implemented the use of 3 and 6 hours had the best performance.

One important generalization note is the use of the control index date. Albeit our trials 2, 3, and 5 appear to have the highest AUC and accuracy scores, these were based on a control index date of the last hour of the icu stay. The use of the last hour may not be realistic for generalization purposes, as our use of a twelve hour observation window likely reflects towards the backend of stay for many patients. Thus, we also included trial 4 that includes an observation window up to the average onset of Sepsis-3 diagnosis found in our case population (approximately .25 LOS). Our results in trial four indicated lower accuracy and AUC scores. Finally, although trials one and four may have lower accuracy and auc scores, we may in

fact desire the use of these models as they provide us with higher sensitivity assuming specificity is at an acceptable level. The cost of a missed Sepsis-3 diagnosis can plausibly outweigh a false positive.

Across our trials, our ROC score is within range of the primary model that we based our study upon Fagerström [3] of .83. Our optimal roc performance is also within range of Kam [9] of .92. Perhaps more importantly, the implications of different observation and prediction windows leads to different sensitivity and specificity results. Overall, our sensitivity and specificity results are high and addresses the notable downside of prior predictive models involving Sepsis-1 and Sepsis-2 criteria.

Conclusion

This study examines using features that are commonly available in Electronic Health Record MIMIC-III database to predict whether a patient may develop Sepsis-3 and specifically provides us the ability to predict its onset. The predictors for the model relies on features identified in Henry et al's [5] studies, and our study adopted a LSTM model [3] to examine if earlier and more accurate results could be achieved.

We examined a set of feature inputs into our LSTM architecture whereby we varied the observation window, prediction window, and control index date. We found strong predictive results across these trials with trials having different resulting strengths dependent on the prediction, observation and control index date utilized.

Team Contributions

All members of our team contributed to the project equally.

References

[1] Z. Chen, M. Pang, Z. Zhao, S. Li, R. Miao, Y. Zhang, X. Feng, X. Feng, Y. Zhang, M. Duan, L. Huang, and F. Zhou, “Feature selection may improve deep neural networks for the bioinformatics problems,” Bioinformatics, 2019.

[2] T. Desautels, J. Calvert, J. Hoffman, M. Jay, Y. Kerem, L. Shieh, D. Shimabukuro, U. Chettipally, M. D. Feldman, C. Barton, D. J. Wales, and R. Das, “Prediction of Sepsis in the Intensive Care Unit With Minimal Electronic Health Record Data: A Machine Learning Approach,” JMIR Medical Informatics, vol. 4, no. 3, 2016.

[3] J. Fagerström, M. Bång, D. Wilhelms, and M. S. Chew, “LiSep LSTM: A Machine Learning Algorithm for Early Detection of Septic Shock,” Scientific Reports, vol. 9, no. 1, 2019.

[4] H. Harutyunyan, H. Khachatrian, D. C. Kale, G. V. Steeg, and A. Galstyan, “Multitask learning and benchmarking with clinical time series data,” Scientific Data, vol. 6, no. 1, 2019.

[5] K. E. Henry, D. N. Hager, P. J. Pronovost, and S. Saria, “A targeted real-time early warning score (TREWScore) for septic shock,” Science Translational Medicine, vol. 7, no. 299, 2015.

[6] Johnson AE, Pollard TJ, Shen L, Lehman LH, Feng M, Ghassemi M, et al. MIMIC-III, a freely accessible critical care database. Sci Data 2016;3:160035 [doi: 10.1038/sdata.2016.35]

[7] Alistair Johnson, and Tom Pollard. "sepsis3-mimic." (2018).

[8] D. A. Kaji, J. R. Zech, J. S. Kim, S. K. Cho, N. S. Dangayach, A. B. Costa, and E. K. Oermann, “An attention based deep learning model of clinical events in the intensive care unit,” Plos One, vol. 14, no. 2, 2019.

[9] H. J. Kam and H. Y. Kim, “Learning representations for the early detection of sepsis with deep neural networks,” Computers in Biology and Medicine, vol. 89, pp. 248–255, 2017.

[10] R. Liu, J. L. Greenstein, S. J. Granite, J. C. Fackler, M. M. Bembea, S. V. Sarma, and R. L. Winslow, “Data-driven discovery of a novel sepsis pre-shock state predicts impending septic shock in the ICU,” Scientific Reports, vol. 9, no. 1, 2019.

[11] Physiobank, physiotoolkit, and physionet components of a new research resource for complex physiologic signals. Goldberger AL, Amaral LAN, Glass L, Hausdorff JM, Ivanov P, Mark RG, Mietus JE, Moody GB, Peng C, and Stanley HE. Circulation. 101(23), pe215–e220. 2000.

[12] D. Zhang, C. Yin, K. M. Hunold, X. Jiang, J. M. Caterino, and P. Zhang, “An Interpretable Deep Learning Model for Early Prediction of Sepsis in the Emergency Department,” 2020.

Appendix A:

Category	Feature Name	Description	Physionet Table
Sepsis Diagnosis	SOFA + Suspected Infection Time	48 hour pre and post window (max sofa score) of Suspected Infection Time	Pivoted_sofa
Vital	Neurologic SOFA	Neurologic SOFA score computed on the basis of GCS	Calculated (with derived sofa)
Vital	FiO2	Fraction of inspired oxygen	Pivoted_fio2
Vital	HR	Heart rate	Pivoted_vital
Vital	RR	Respiratory rate	Pivoted_vital
Vital	GCS	Glasgow coma score	Pivoted_gcs
Vital	SBP	Systolic blood pressure	Pivoted_vital
Vital	Shock Index	HR/SBP	Calculated (with vitals)
Laboratory	Bun/CR	BUN/creatinine ratio	Calculated (with labevents)
Laboratory	Arterial pH	Blood pH as measured by an arterial line	Pivoted_bg
Laboratory	PaO2	Partial pressure of arterial oxygen	Pivoted_bg
Laboratory	BUN	Blood urea nitrogen	Pivoted_lab
Laboratory	Hepatic SOFA	Hepatic SOFA score computed based on the bilirubin concentration	Pivoted_sofa (liver)
Laboratory	Renal SOFA	Renal SOFA score computed on basis of creatinine concentration	Pivoted_sofa (renal)
Laboratory	WBC	White blood cell count	Pivoted_lab
Laboratory	Platelets	Platelet count in the bloodstream	Pivoted_lab
Clinical	Avg Urine	Total urine output over the past 6 hours	Calculated (with urine_output.sql)
Clinical	Chronic Liver disease and cirrhosis	Presence of chronic liver disease and cirrhosis as indicated by ICD-9 571	Calculated (with diagnoses_icd)
Clinical	Cardiac surgery patient	Patient currently in the cardiac surgery recovery unit	Calculated (with diagnoses_icd)

Clinical	Immunocompromised	Immunocompromised (patient has received past therapy that suppresses resistance to infection) as indicated by presence of any ICD-9 in V58.65, V58.0, V58.1, 042, 208.0, 202	Calculated (with diagnoses_icd)
Clinical	Hematological Malignancy	Presence of hematologic malignancy as indicated by any ICD-9 code in 200-208	Calculated (with diagnoses_icd)
Clinical	SIRS	Presence of at least two of the SIRS criteria at the current time	Calculated (with vitals_first_day, lab_first_day, blood_gas_first_day_ arterial)
Clinical	Chronic heart failure	Presence of heart failure as indicated by ICD-9 code 428	Calculated (with diagnoses_icd)
Clinical	Chronic organ insufficiency	Severe organ insufficiency (chronic liver disease, chronic heart failure, chronic respiratory failure, receiving chronic dialysis) as indicated by one of the ICD-9 codes 571, 585.6, 428.22, 428.32, 428.42, 518.83	Calculated (with diagnoses_icd)
Clinical	Diabetes	Presence of diabetes as indicated by ICD-9 code 250	Calculated (with diagnoses_icd)
Clinical	Metastatic carcinoma	Metastatic carcinoma as indicated by presence of any ICD-9 codes in 140-165, 170-175, 179-199	Calculated (with diagnoses_icd)

***Red Highlight denotes features that were not replicated**