Comparing NLP Methods to Understand Clinical Text to Improve Outcomes in Sepsis Patients

Stanford CS224N Custom Project

Gowri Nayar
Department of Computer Science
Stanford University
gnayar@stanford.edu

Abstract

Patients that will develop sepsis often present within the Emergency Department with a quick progression of the infection. In order to identify patients at risk for sepsis, the common methodology relies on the calculation of a SOFA score, that depends on vitals measurements that are taken as the patient remains in the hospital. However, this score is not always accurate and can also be too slow, as vital measurements get updated within the electronic health system slowly. Therefore, we investigate a method based on the classification of nurse triage notes, that can identify the risk for developing sepsis. From this analysis, we can see that the paragraph vectors outperform the bag-of-words embedding models in accuracy of prediction, achieving an accuracy of 0.95 and an AUC of 0.66.

1 Key Information to include

- Mentor: Gaurab Banerjee
- External Collaborators (if you have any): NA
- Sharing project: NA

2 Introduction

Sepsis is an illness related to the immune system in response to an infection, and can escalate quickly in severity causing a serious risk of death and organ damage. While there are existing treatments that effectively reduce the symptoms of sepsis, identifying the condition is crucial in preventing long-term damage. This presents a challenge particularly in emergency situations, as the clinical staff, nurses and doctors, are often over-loaded with information and cannot make a clear diagnosis. The Surviving Sepsis Campaign has implemented protocols for identifying sepsis early, but these protocols rely on vital measurements which are slow to get inputted into electronic health records (EHR), and thus are slow to monitor. Lastly, a sepsis diagnosis requires an integration of data from a variety of sub-groups of the healthcare system, including imaging, vitals, and patient observation, which can be difficult to aggregate.

The increase in EHR data has allows machine learning to tackle many problems that are present within the clinical setting. However, most applications are focused on the structured data contained with the EHR dataset, including vitals measures. While it has been shown that performing logistic regression over these structure datatypes, such as blood pressure, heart rate, and white blood cell count, can increase the accuracy of sepsis prediction than sole clinician diagnosis [1, 2], we can also look to the clinical text that is stored with every entry into the EHR to make better, more timely predictions.

This work focuses on using these clinical notes to perform an embedding and classification of the notes to predict whether the patient will develop sepsis, severe sepsis, septic shock, or be discharged. There

has been a dramatic increase in developing NLP methods that are specific to domain information, so we will compare two methods of embedding the free-form text, one with a simple bag of words model and the second with an embedding layer. We will show that both of these methods increases the accuracy of prediction and discuss key variations within each model when learning on the clinical text.

3 Related Work

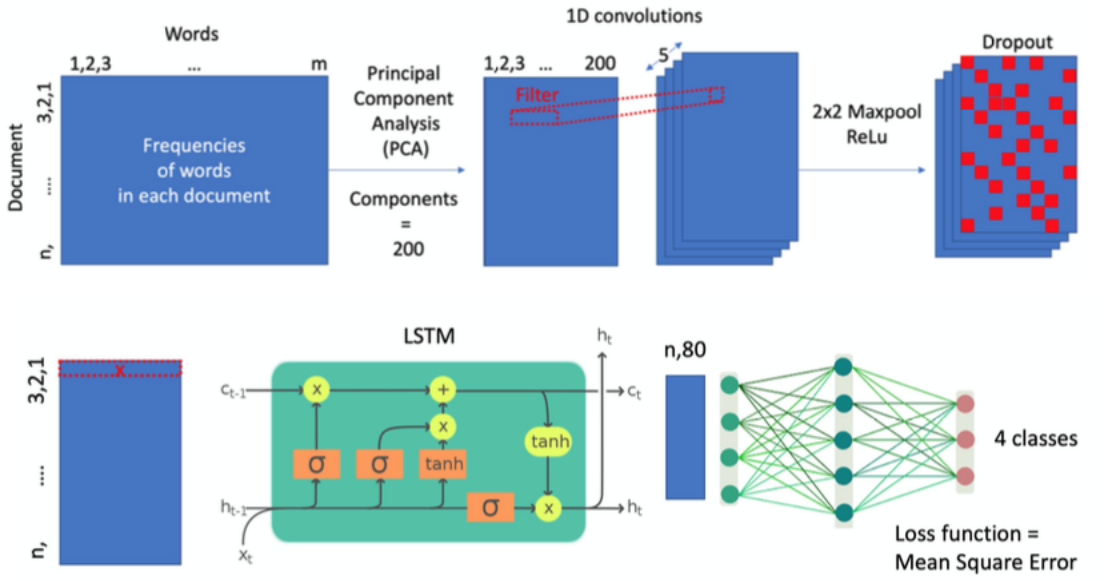
This work draw inspiration from a recent publication by Sterling et al. who describe a methodology to predict emergency department outcomes using NLP methods on triage notes. [3] This paper reports to be the first to apply specific NLP techniques to nursing triage notes and generate a predictive model. Specifically, the main contribution is in using paragraph vectors to create numeric representations of phrases that allow for semantic comparisons with other phrases within the collection of documents. The paragraph vector methods allows the preservation of word ordering and negations, and so the authors hypothesize that it will perform better than a bag-of-word approach in the prediction model. Furthermore, this paper utilizes topic modeling as well to model the content across a set of documents, where each document represents a mixture of topics. This paper claims novelty in using these methods on nurse ER triage text in order to develop a prediction model that will determine the eventual needs of the patient, ie hospitalization or discharge.

This study conducted was a retrospective study and this poses a limitation on the analysis conducted. For instance, there is the risk of the outcome in the medical record to be incorrect, and biases within the nurses notes. There is no real-time impact analysis of the prediction model on ED operation, also because of the retrospective nature, and so the authors cannot test whether their model impacted the workflow of the ED. Furthermore, the NLP embeddings should also take into account the 'sentiment' of clinical notes, and develop a method to normalize across the documents within the set. Even with these limitations, this paper shows promise in using only the textual notes from triage to evaluate a patient's status.

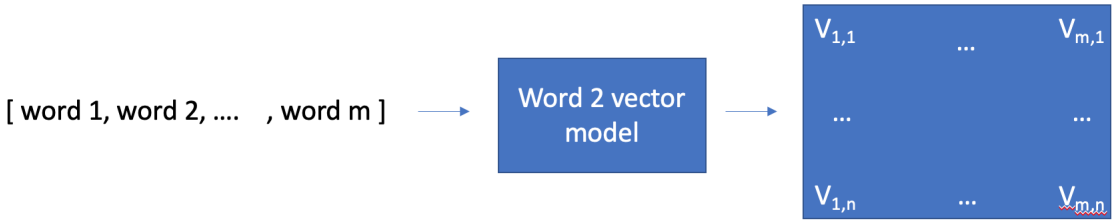
4 Approach

The task of this project is to develop a representation model for EHR triage notes to then train and test a predictive model that will determine the patient outcome (discharge, sepsis, sever sepsis, or septic shock). This is a retrospective study, as I use the eventual outcome of the patient as the label. First, I aggregate the data from the MIMIC database, and curate this data to only include patients that are given triage notes within the first 12 hours upon arrival. We only include patients that are either discharged or admitted into the ICU, as these are the only two cohorts we are concerned about in terms of developing sepsis. I first perform a text preprocessing to remove stop words (such as 'is', 'that', 'and'), punctuation, and numbers, and then use this as input for two different methods of embedding. This data and preprocessing step is further described in the data section below.

This problem is first viewed as an embedding problem of the textual data and then a classification model from the resulting embedding. First, I implement a simple bag-of-words model that tokenizes each word and maintains a matrix representing frequency. I perform PCA and maintain the top 200 principle components as input for the classification model. This method first utilizes layers of CNN to perform the classification. This is to allow for the model to quickly reduce the dimensionality of the large feature matrix and place a higher weight on the features (or words) that are the most discriminant. [4] Then, the output of the CNN is inputted into a layer of LSTM, as is common in text classification purposes. Lastly, the output of the LSTM is used with a densly connected layer that uses a softmax activation function to create a multi-class probability for each note, where each class denotes either discharge, sepsis, severe sepsis, or septic shock. Post-processing on the output vector is done to identify the class with the highest probability and assign this class label to the note. The structure of this base model is denoted in Figure 1.



The second method first uses an embedding layer, that is created through a set of word vectors that are trained on the corpus of words that are in the total set of notes. Each word in the training set of clinical note is denotes as a vector, using the Word2Vec architecture [5]. This set of vectors becomes the input to the embedding layer at the start of the network. This serves to tune the model to the specific set of words that are relevant to this problem space. The vectorized notes are then used as input into the same base model as described above. The structure of this model with the embedding layer can be found in Figure 2. The exact configurations for each run can be found in the Experiments section below.



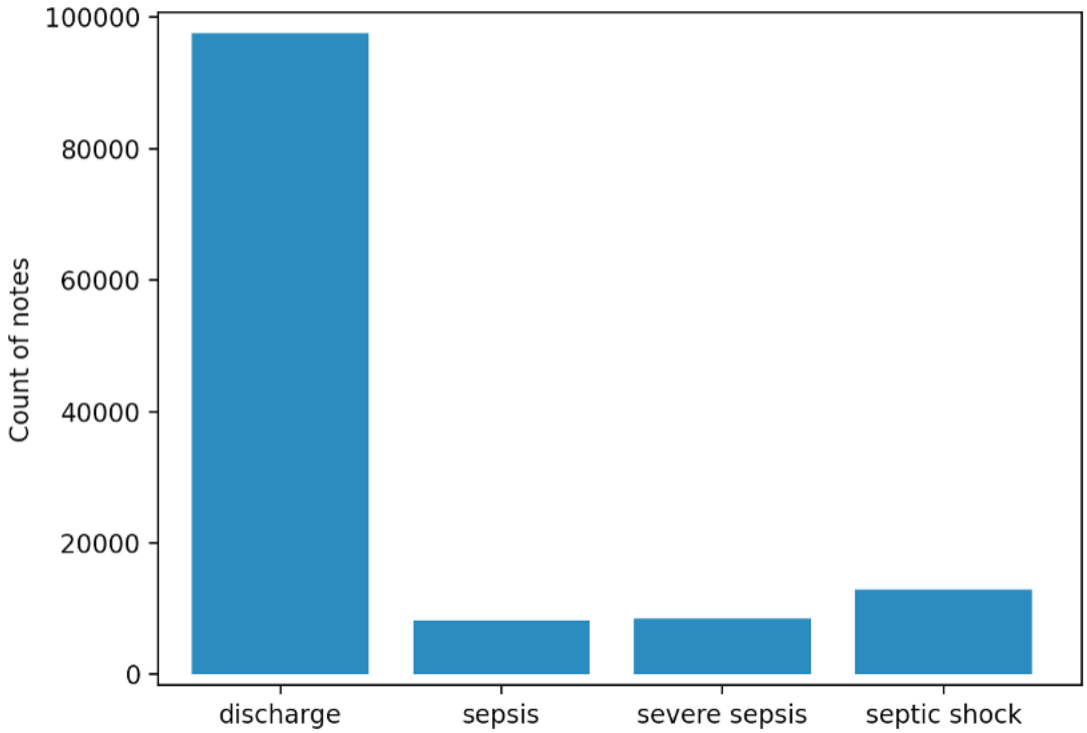
As baseline, the accuracy of these models are compared to the metrics published by Sterling et al, as they describe an AUC of 0.72 and and F1 score of 0.57, and accuracy of the SOFA clinical-based method of 0.75. [6]

5 Experiments

5.1 Data

The study uses data from the MIMIC-III dataset, which contains anonymized EHR data from over 50,000 admissions. Patients that were diagnosed within the first 3 hours were excluded, because their clinical notes do not fall into the category that is needed for this sepsis study. The resulting number of 127126 clinical notes, with a high bias towards notes that correspond to a non-sepsis

related diagnosis. There are a total of 29616 sepsis-related notes and 97510 non-sepsis related notes. The full break-down of each note type (sepsis, severe sepsis, septic shock) is shown in Figure 3.



In order to account for the bias in data, upsampling the sepsis-related notes is used. When creating the training set, the data is augmented with sepsis-related notes, adding 20% of sepsis-related notes to the training set. A sepsis related note is chosen with uniform probability, and it is perturbed to add noise to each of the vector values, before being added to the training dataset. While this reduces the effect of bias in the model training, it does introduce other errors, as the same sample is trained upon multiple times. This is discussed further in the Analysis section below.

5.2 Evaluation method

Two of the primary metrics used are the accuracy and the roc/auc metrics. Accuracy is calculated by taking the percentage of the correctly predicted values to the total number of predictions. The ROC curve is calculated with the x-axis representing the $1-$specificity $= FP/(FP + TN)$), where FP is false positive and TN is true negatives. The y-axis is the sensistivity $= TP/(TP + FN)$. The AUC metric is the area under the ROC curve, which represents an aggregate measurement across all classification thresholds. We also use an F1 metric in order to compare to the baseline metrics described in the comparison paper. $F1 = 2 * ((precision * recall)/(precision + recall))$, where precision $= TP/TP+FP$ and recall $= TP/TP+FN$. The F1 metric conveys the balance between precision and recall for the prediction.

5.3 Experimental details

For the both models, a learning rate of 0.001 is used. The models were implemented using tensorflow and keras. For the bag of words model, a randomized method is used when calling PCA, as is typical for high dimensionality data. For the Word2Vec embedding, words the occur less than 10 times are removed, with a window size of 5. The layers of CNN are trained with a learning rate of 0.001, the number of filters are 100, and the activation function is a ReLU. The LSTM uses 80 units and a tanh activation function. The densly connected layer uses a softmax activation function. The models are

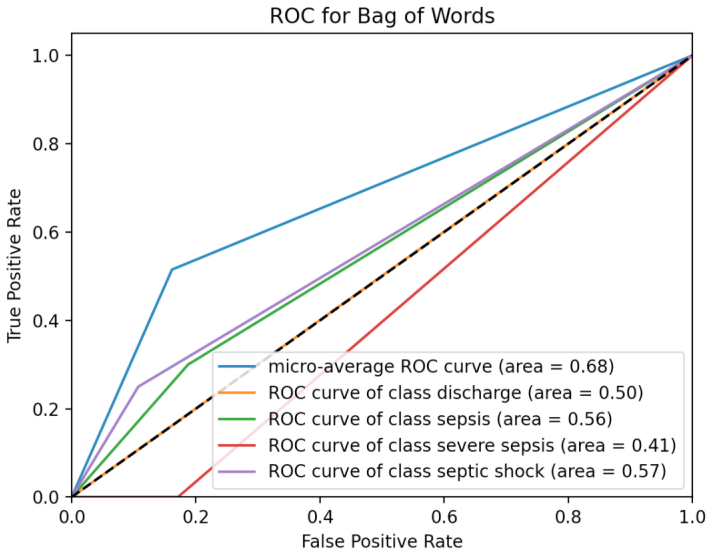
trained with 150 epochs, with early stopping, and a batch size of 128. The models are optimized using an adam optimizer. For the emedding layer in the second network, the input is the entire size of the corpus and the output is 200, to be comparable to the matrix outputted from the PCA in the first model. Dropout layers of 0.2 are also used.

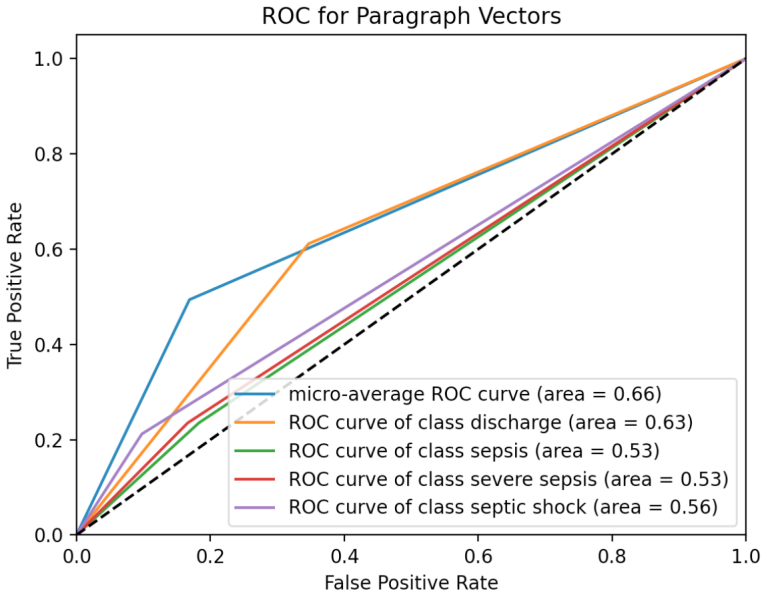
Both models take about 10 minutes to train. This training time can be attributed to the dimensionality reduction steps performed at the beginning of each model. Below are the final epochs for each model, showing the loss and accuracy at the early stopping point:

```
Epoch 63/150
6/6 [==============================] - 3s 442ms/step - loss: 0.0324 - accuracy: 0.9328
Epoch 63/150
6/6 [==============================] - 2s 257ms/step - loss: 0.0178 - accuracy: 0.9597
```

5.4 Results

The following two graphs show the ROC and AUC for the bag of words and paragraph vector methods.





The AUC are 0.68 and 0.66 for bag of words and the paragraph vectors respectively. These overall AUC number are as expected for such a dataset with 4 class and a high-dimensionality, however a higher AUC for the embedding method would be expected. The decrease in performance may be attributed to the upsampling performed to account for the bias in data, as the netowrk is trained multiple times on the same word vectorization, and thus the entire vector space is under-utilized.

Model	Accuracy	AUC	Recall	Precision	F1
Baseline	0.725	0.63	0.693	0.736	0.572
BOW	0.93	0.68	0.588	0.14	0.23
Paragraph Vector	0.95	0.66	0.29	0.15	0.2

The F1 scores are lower than the baseline model, possibly due to the bias in data.

6 Analysis

The two models perform well on recall, particularly when there are specific words that can be attributed with a particular class. For example, within the septic shock class, the most extreme diagnosis, there are particular words used, such as fatal, intense, confusion, which are not used within other classes. This could explain the higher performance within this class.

The dimensionality reduction performed at the beginning of each model serves to drastically reduce the training time required, but also impacts the precision and recall of the models. By decreasing the dimension of the input matrix, the models could be hyperfitting to the data provided, and thus have a lower performance on new data. This could also be affected by the upsampling performed during the pre-processing step. This causes the model to be trained on the same example multiple times, though perturbed, which can again cause hyperfitting.

The overall performance of the paragraph vectors model is higher, as the embedding is important in the classification of text within a domain with specific terminology, as in the clinical setting. The overall performance could reflect the bias and noise in the data, as more tuning would be able to further remove words that are not differentiating. Upsampling sepsis samples would impact the embedding as they are not distributed across the vector space

7 Conclusion

This work ultimately shows the usefulness of free-from clinical text, which was previously excluded from automation systems. Both methods provide an improvement in accuracy in determining the outcome of a patient from standard clinical methods. However, some limitations are shown through the F1 score which could be improved through different training methods involving transformers. This would account for the large feature vector for each note and create a more precise model.

References

[1] Fatemeh Amrollahi, Supreeth P. Shashikumar, Fereshteh Razmi, and Shamim Nemati. Contextual embeddings from clinical notes improves prediction of sepsis. *AMIA Annu Symp Proc.*, 2021.

[2] Simon Meyer Lauritsen, Mads Ellersgaard Kalør, Emil Lund Kongsgaard, Katrine Meyer Lauritsen, Marianne Johansson Jørgensen, Jeppe Lange, and Bo Thiesson. Early detection of sepsis utilizing deep learning on electronic health record event sequences. *Artificial Intelligence in Medicine*, 104:101820, 2020.

[3] Nicholas W. Sterling, Rachel E. Patzer, Mengyu Di, and Justin D. Schrager. Prediction of emergency department patient disposition based on natural language processing of triage notes. *International Journal of Medical Informatics*, 129:184–188, 2019.

[4] Gaowei Xu, Tianhe Ren, Yu Chen, and Wenliang Che. A one-dimensional cnn-lstm model for epileptic seizure recognition using eeg signal analysis. *Front. Neurosci*, 2020.

[5] Tomas Mikolov, Kai Chen, G.s Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *Proceedings of Workshop at ICLR*, 2013, 01 2013.

[6] Alan Jones, Stephen Trzeciak, and Jeffrey Kline. The sequential organ failure assessment score for predicting outcome in patients with severe sepsis and evidence of hypoperfusion at the time of emergency department presentation. *Critical care medicine*, 37,5:1649–54, 2009.