Portable Early Prediction of Sepsis from Clinical Data on Intel Myriad X

June-August 2019

AUTHOR(S):
Priyanka Kumar Mathur

CERN openlabs

SUPERVISOR(S):
Alberto Di Meglio
Sofia Vallecorsa
Taghi Aliyev
Fabio Fracas

CERN openlab

PROJECT SPECIFICATION

CERN openlab has recently started an investigation to design and implement a large-scale distributed platform to perform data analysis of medical and other personal protected data for research and clinical purposes, called CERN livinglab. The goal of CERN openlab is to investigate the technological challenges generated by such a platform and work with medical researchers, biologists and doctors to collect requirements and use cases and to validate the early prototypes of the platform.

The CERN livinglab team is interested in evaluating state-of-the-art tools and technology and build a growing set of functionality, methods and best practices to be integrated in the platform. In this context we are proposing a project to work on the PhysioNet/Computing in Cardiology Challenge 2019, whose focus is “Early Prediction of Sepsis from Clinical Data”. The developed deep learning model will be loaded on a laptop computer or tablet equipped with the Myriad USB key and able to provide rapid predictions of sepsis probability in clinical scenarios.

ABSTRACT

Sepsis is a life-threatening condition where microbes present in the blood stream cause an unregulated immune response from the body which can result in tissue damage, multi-organ failure and eventually death. It affects 30 million people worldwide and causes 6 million deaths. Studies have indicated that every hour sepsis goes undetected, patient mortality increases 4-8%, thus early detection of the disease is necessary to decrease mortality rates and provide better patient outcomes.

This project is derived from this year's 2019 Computing in Cardiology Challenges, which focuses on the early detection of sepsis using machine learning algorithms. The two primary focuses of the project include (1) developing a model which can successfully detect sepsis early and (2) demonstrating the clinical viability of this deep learning model by implementing the model on Intel's Neural Compute Stick 2 (Myriad X VPU processor) which is a portable USB device that can implement and deploy deep learning models.

While developing the model, the data was preprocessed in the following ways: (1) linear interpolation was used to fill all missing values and values not collected were 0, (2) synthetic minority oversample technique (SMOT) was used to balance the data set, (3) patient file lengths were standardized to have only 3 and 17 hours of patient data, (4) all 40 parameters were used (although a variety of subsets were tested). Data preprocessing was done using MATLAB.

The model was developed in python using Tensorflow and Keras. The model itself was a seven layer dense neural network (DNN) with Leaky Rectified Linear Unit (ReLU) activation functions for all hidden layers and a sigmoid activation function for the output layer. This was paired with binary cross entropy loss.

The resulting model detected sepsis well, with a 76% sensitivity (correctly identified sepsis) and 80% specificity (correctly identified healthy patients). The model was successfully implemented on Intel's Neural compute stick 2 (Myriad X VPU processor) with an average of 10ms processing time to determine whether the patient had sepsis, demonstrating clinical viability. Further testing including tuning hyperparameters to further increase sensitivity and specificity is currently being done.

TABLE OF CONTENTS

1. Introduction

a. Machine Learning in Healthcare

Healthcare and medicine have historically remained independent of computing advances, only recently integrating paper records to patient electronic health records (EHRs); however, with the rise of computing power as well as the scalability of this computing power at low costs, large quantities of information found in medical health records (estimates of 1 trillion gigabytes in the United States alone), provides an opportunity for the use of computing, specifically machine learning and artificial intelligence, to process healthcare data for early disease detection, drug discovery and precision medicine.

One promising area of research in machine learning in healthcare is its use in early detection of diseases. Early detection of diseases is critical as it increases treatment options for patients as well as decreases mortality rate to provide better patient outcomes. Scientists have continually sought ways to detect diseases early, usually through cellular pathology, biomarkers and genetic tests. Perhaps one of the most famous examples of early disease detection was the invention of the Pap smear, which examines the pathology of cells for cervical cancer detection. With this early detection and screening methods, mortality rate of cervical cancer dropped over 50% (Safaeian, & Solomon, 2009). Other well-known early detection tests for diseases include those for breast cancer detection of BRCA1 and BRCA2 genes. These traditional methods of early detection utilize genetic screenings, biomarker detection or protein detection in bodily fluids; however, given the wealth of information found in electronic patient health records, using deep learning and AI on electronic patient health records could allow for early disease detection. Using deep learning early detection could improve patient outcomes and be applied to a broad range of diseases.

Many recent studies have been using machine learning in healthcare and diagnostics. A recent study indicated that machine learning algorithms were able to detect Alzheimer's six years earlier than doctors were able to (Ding et al., 2019). The complex and subtle changes in brain structure on imaging scans associated with early stages of a disease are often too subtle for doctors to analyze. However, the large computing power of machine learning allows for it to detect and process these changes, finding viable diagnosis earlier than doctors are able too. Other studies have also been able to detect cancerous tumors as well as analyze clinical radiographic images more accurately and faster than physicians are able to (Haenssle et al., 2018; Abramoff, 2018).

Currently, the most successful machine learning algorithms for disease detection use image analysis of radiographical scans (PET, SPECT, CT, MRI etc.), which often use convolutional neural networks to analyze the images (Soffer et al., 2019). However, many diseases are not detectable via image scans and bloodwork analysis can take a while. Thus, it is necessary to utilize electronic patient health records (EHRs) to diagnosis diseases. EHRs contain a plethora of information including lab work, vital signs and demographic information which can be carefully processed and inputted into a deep learning algorithms for disease detection. One such condition that cannot be detected from image analysis and requires bloodwork is sepsis.

b. Sepsis

Sepsis is a condition in which microorganisms enter the bloodstream and the body has an unregulated immune response to the presence of these microorganisms which can result in tissue damage, organ failure or death (Singer et al., 2016). In the US approximately 1.7 million people are estimated to develop sepsis with approximately 270,000 people dying from it yearly (CDC, 2016). It is estimated that approximately one in three deaths in US hospitals are associated with sepsis (CDC, 2016). Internationally, sepsis affects nearly 30 million people worldwide with 6 million deaths each year (WHO, 2018). Furthermore, approximately 4.2 million infants are afflicted with sepsis (WHO, 2018). In addition to patient care, sepsis has large economic costs. In the US alone, sepsis costs US hospitals approximately 24 billion dollars per year (13% of US healthcare costs), more than any other disease;

the majority of those cases were patients who were not diagnosed with sepsis upon admission (Paoli et al., 2018).

Furthermore, sepsis is a time sensitive disease and is extremely fast acting. It is estimated that for every hour sepsis goes undiagnosed the mortality rate increases 4-8% (Kumar et al., 2006; Seymour et al., 2017). While clinicians have developed new guidelines for sepsis diagnosis to combat late detection, other methods for early detection and predication of sepsis could provide better treatment options. The most common method of sepsis diagnosis defines a patient's state of sepsis as a two point increase in their sequential organ failure assessment (SOFA) score (Singer et al., 2016). This methodology often results in delayed detection/treatment. Although new guidelines to detect sepsis have been outlined to decrease detection time, including quick sequential organ failure assessment (qSOFA) scores, earlier prediction of sepsis is still necessary to decrease patient mortality.

Early detection of sepsis shows promise as the best way to decrease mortality of the disease. Using time series data sets of EHRs that contain vital signs, demographic information and lab work, machine learning algorithms have promise for early detection of sepsis.

c. PhysioNet Computing in Cardiology Challenge 2019

This year's computing in cardiology challenge utilizes machine learning for the early prediction of sepsis using physiological time-series data from time of admission to the ICU. For this challenge, sepsis was defined according the Sepsis-3 guidelines which dictates a 2-point change in the patient's sequential organ failure assessment and clinical suspicion of infection (ordering blood cultures or IV antibiotics). The goal of this challenge was to predict sepsis earlier than clinicians were able to detect it (Reyna et al., 2019). The data set for each patient contained 40 parameters including vital signs, laboratory results and demographic information with hourly updates to the information (missing data is in the set as not all values were collected/recorded every hour) (Reyna et al., 2019). This project will be based on this year's 2019 Computing in Cardiology challenge (although slightly modified), and will focus on utilizing deep learning models for the early detection of sepsis.

d. Intel's Myriad X VPU and the Neural Compute Stick 2

Using AI to predict diseases early is very important; however, it requires a practical method to implement within hospitals which includes speed, portability and low cost. Intel's Neural compute stick 2, which has the Myriad X Vision Processing Unit (VPU) is a portable USB connection device which can implement and deploy deep learning algorithms. It's portability, speed and low cost make it a practical device to implement deep learning models for early detection of diseases and its application will be further explored in this project.

2. Project Goals

This project is based on the Computing in Cardiology 2019 Challenge which focuses on the early prediction of sepsis using machine learning models. There are two primary goals for this project including the development of a neural network and practical implementation of that network for the early detection of sepsis.

The first goal for this project is to develop a working deep learning model to predict sepsis earlier than clinicians are able too. This model should utilize the time series data provided by the 2019 Computing in Cardiology Challenge to predict sepsis with greater than 70% sensitivity (true positive) and specificity (true negative). Ideally the model should be able to predict sepsis with greater than 80% specificity and sensitivity for more practical clinical aided diagnostics. The model should also be able to predict sepsis more than 10 hours before clinicians are able to.

Additionally, for this model appropriate parameters from the dataset should be selected. Furthermore, the model should be a dense neural network and not a long-short term memory network, as the neural compute stick does not have support for these models.

The second goal of this project was to implement the model on Intel's Neural Compute Stick 2 and measure its clinical viability by determining the speed at which the model could predict each patient's case as well as cost of the device and implementation in the hospital.

The first goal of this project, development of a neural network, has many parts including: (1) selecting parameters available, (2) examining the dataset (3) pre-processing the dataset and (4) developing a working model as well as tuning the hyperparameters of the model.

3. Neural Network

a. The Dataset

The 2019 Computing in Cardiology PhysioNet Challenge, provided 2 publically available datasets, one for testing and one for validation, which each contained approximately 20,000 patient files. Each patient file consisted of 40 patient parameters listed in the tables below:

Vital signs (columns 1-8)	
HR	Heart rate (beats per minute)
O2Sat	Pulse oximetry (%)
Temp	Temperature (Deg C)
SBP	Systolic BP (mm Hg)
MAP	Mean arterial pressure (mm Hg)
DBP	Diastolic BP (mm Hg)
Resp	Respiration rate (breaths per minute)
EtCO2	End tidal carbon dioxide (mm Hg)
Laboratory values (columns 9-34)	
BaseExcess	Measure of excess bicarbonate (mmol/L)
HCO3	Bicarbonate (mmol/L)
FiO2	Fraction of inspired oxygen (%)
pH	N/A
PaCO2	Partial pressure of carbon dioxide from arterial blood (mm Hg)
SaO2	Oxygen saturation from arterial blood (%)
AST	Aspartate transaminase (IU/L)
BUN	Blood urea nitrogen (mg/dL)
Alkalinephos	Alkaline phosphatase (IU/L)
Calcium	(mg/dL)
Chloride	(mmol/L)

Creatinine	(mg/dL)
Bilirubin_direct	Bilirubin direct (mg/dL)
Glucose	Serum glucose (mg/dL)
Lactate	Lactic acid (mg/dL)
Magnesium	(mmol/dL)
Phosphate	(mg/dL)
Potassium	(mmol/L)
Bilirubin_total	Total bilirubin (mg/dL)
TroponinI	Troponin I (ng/mL)
Hct	Hematocrit (%)
Hgb	Hemoglobin (g/dL)
PTT	partial thromboplastin time (seconds)
WBC	Leukocyte count (count*10^3/µL)
Fibrinogen	(mg/dL)
Platelets	(count*10^3/µL)
Demographics (columns 35-40)	
Age	Years (100 for patients 90 or above)
Gender	Female (0) or Male (1)
Unit1	Administrative identifier for ICU unit (MICU)
Unit2	Administrative identifier for ICU unit (SICU)
HospAdmTime	Hours between hospital admit and ICU admit
ICULOS	ICU length-of-stay (hours since ICU admit)

Furthermore, of the 20,000 patient files only about 5% were cases of sepsis, creating an imbalanced dataset. Additionally, while evaluating the parameters, it was noted for each parameter anywhere between 20-80% of patient data was missing. This can be seen in **Figure 1** below which shows the frequency of the O_2 saturation parameter per hour of patient stay. O_2 saturation is considered a vital sign parameter and is thus recorded with regular frequency. However, it is noted that even with this parameter at least 10% of patient data is missing at any point in time. Additionally, looking at parameters that are taken less often such as those from lab tests including white blood cell count (WBC) the amount of missing data increases even more, with 80-90% of data missing per hour as seen **in Figure 2**.

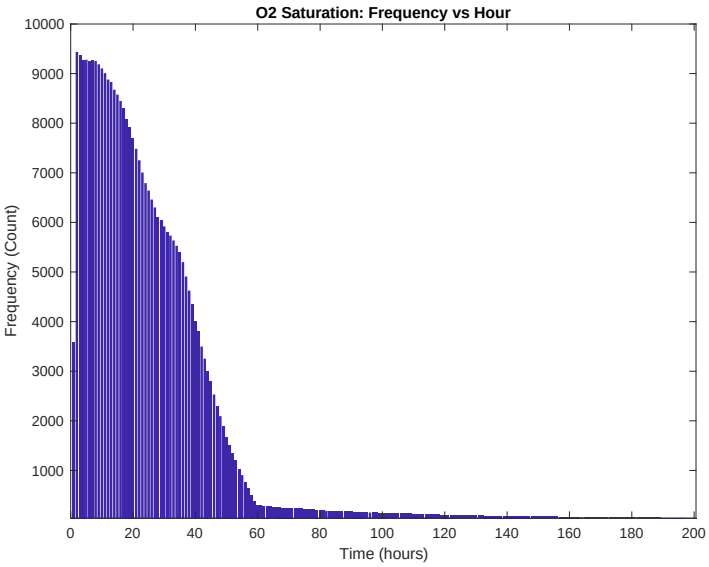

Figure 1: Frequency of O2 saturation in patient data parameter per hour

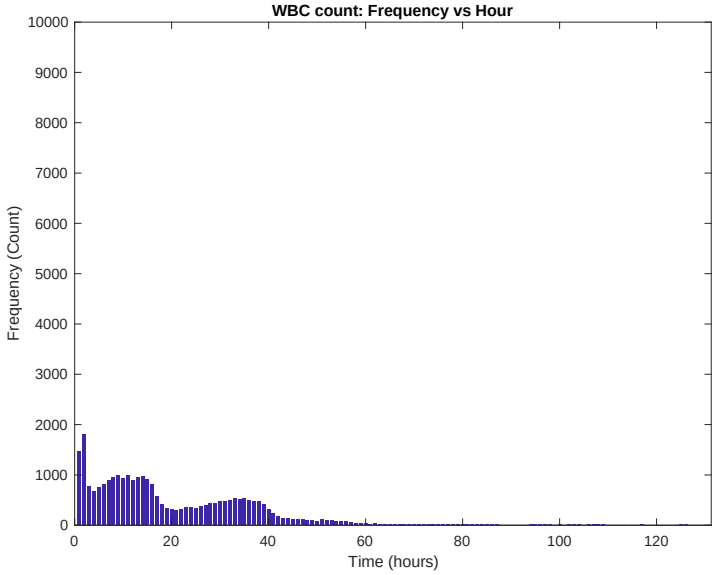

Figure 2: Frequency of white blood cell count in patient data parameter per hour.

This dataset has several features worth noting which make data preprocessing more complex. Firstly, there are a large amount of parameters (40), a significantly imbalanced dataset, lots of missing data and varying lengths of patient stay. All of these challenges are often found in medical datasets. The wealth of information from EHRs is reflected in large amount of parameters. The imbalanced dataset is indicative of the relatively low prevalence of the disease within a standard patient population. Missing data is expected due to the infrequency of lab work (usually not done every hour) as well as the fact that parameters are usually not taken every hour (however, vital sign parameters are taken with relatively high frequency compared to those of lab test results). Finally, patient stay lengths vary due to diagnosis and time of release. These are all challenges found with medical datasets which will be addressed in data preprocessing.

b. Data Pre-Processing

The data pre-processing was done in MATLAB. Data preprocessing reflected the challenges found within the medical data set. Four primary challenges found within the medical dataset were: (1) large numbers of parameters, (2) imbalanced dataset, (3) missing data, (4) varying lengths of patient stays.

i. Large Number of Parameters

To address the large number of parameters, three modified versions of the data set were selected with varying parameters. The first dataset contained all 40 parameters. The second dataset contained only 8 vital sign parameters. Finally, the last data set contained 19 parameters, including the 8 vital sign parameters as well as 11 parameters which have been shown to correlate with sepsis including: fraction of inspired oxygen (FiO2, %), pH, blood urea nitrogen (BUN, mg/dL), Creatinine (mg/dL), direct bilirubin (mg/dL), total bilirubin (mg/dL), serum glucose (mg/dL), lactate (lactic acid, mg/dL), partial thromboplastin time (seconds), white blood cell count (leukocyte count, count*10^3/uL) and platelet count (count*10^3/uL). These additional 11 parameters have been shown to correlate with sepsis, in machine learning studies (Shimabukuro et al.,2017). .

ii. Imbalanced Dataset

Perhaps the most difficult challenge to address was the imbalanced dataset, with only 5% of patients having sepsis. When the data was directly put into the model it yielded a sensitivity of 0%, but specificity of 95%. This resulted as the model only predicted cases of no sepsis (the majority class), thus yielding a falsely high accuracy which had 0% sensitivity to sepsis. Thus, inputting the data for training without preprocessing to balance the majority and minority data would not be possible. The training data preprocessing would require balancing the majority and minority dataset classes.

Several approaches were tried to balance these datasets that did not work including downsampling the majority class, repeating/duplicating cases of the minority class to artificially increase the number of sepsis cases and adding class weights.

Downsampling the majority required using only 1,000 of the majority sample cases such that there was an approximate equal amount of majority and minority class samples. However, the model was still unable to predict cases of sepsis, resulting in only a 10% accuracy in the results. This likely resulted from the fact that 2,000 samples total is not enough data to train the model on, given the large size of the data set. This method worked with none of the three parameter sets.

Duplicating the minority class to increase the number of minority class samples, similarly resulted in the prediction of only one class as well as low accuracy for validation data with less than 50% accuracy (again predicting only one class). This likely resulted from the fact that repetition of minority samples caused the model to “memorize” training data and could not duplicate results on the validation data. Finally, adding class weights also resulted in 0% sensitivity in the results, still only predicting the majority class.

Furthermore, a combination of all the methods as well as a subset of the methods were utilized. Downsampling the majority class as well as replicating those in the minority class yielded similar results in which the model predicted only one class and having an accuracy of 50% and a sensitivity of 0%. Combining either the downsampling or replication of the minority class with class weights, similarly resulted in the model having a 0% sensitivity to sepsis. Combination of all three methods also resulted in 0% sensitivity of the model.

After trying these methods, a method known as synthetic minority oversampling technique (SMOT) was used. SMOT increases the minority class dataset by creating new patients from the minority class without replication. This relies on adding synthetic Gaussian noise to minority class patient data to create new minority class patients without directly replicating the minority class. Gaussian noise with a mean 0 and standard deviation of 0.1 was added to the patient data (other levels of Gaussian noise with mean 0 and standard deviation (sd) 0.01, mean 0.5 and sd 0.1, mean 0.5 and sd 0.01, were added, however, provided the worse specificity and sensitivity). The use of SMOT worked well and resulted in a model with over 75% sensitivity and specificity. In addition to using SMOT, class weights were also used in the model.

iii. Missing Data

Missing data proved to be a large challenge for data preprocessing as well. With anywhere between 10-80% of data missing for any parameter several approaches were tried. The first approach, very standard for missing data, was zero padding the data such that all missing values were set to 0 and -1 (in two different trials). -1 was chosen because negative numbers are not found in medical datasets and would thus be unique to missing data. However, neither of these methodologies yielded good results with accuracy of around 50%. Because this was a time-series dataset, the expected changes in parameter values were expected to change in time. The estimated relationship between time points for a specific parameter was estimated to be linear. Thus missing values for each parameter were filled in with linear interpolation between the data points. Each parameter was then subsequently normalized such that the values ranged only between 0 and 1. If a parameter was not collected it remained as 0.

iv. Varying Length Stays

Inputting various size inputs into a model is not possible, thus the input size must be standardized. It was noted that the average detection time for sepsis was 55 hours and thus the patient files were standardized to approximately that size initially. With this many hours of data, the model was able to predict sepsis with upwards of 80% sensitivity and specificity. However, given that the challenge focused on early sepsis detection, timepoints of 17 hours and 3 hours were given as benchmarks for sepsis detection. For each of the two trials only 3 hours and 17 hours of patient data were inputted into the model. These two timepoints were used as benchmarks in the model for early prediction of sepsis and thus fulfilled the varying length of stay concern, limiting patient data to 3 and 17 hours.

Overall the final data set was preprocessed in the following ways:

(1) To address a large number of parameters, three parameter subsets were created for the model to be tested on: all parameters (40), vital sign parameters (8) and vital sign parameters + optional parameters (19) that have been shown to correlate with sepsis. The one with all parameters (40) was utilized in the final model.

(2) To address an imbalanced dataset, SMOT was used, in which random Gaussian noise was added to minority class patient datasets to artificially create new patients in the minority class (sepsis). The random Gaussian noise had mean 0 and standard deviation 0.1.

(3) To address missing data linear interpolation was used to fill in missing data. Any parameter that was not taken, was set to 0. Additionally, all parameters were normalized relative to itself such that all values of a parameter existed between 0 and 1.

(4) To address the varying lengths of each patient file, two benchmark data timepoints were used for early detection of sepsis: 3 and 17 hours.

c. The Model

i. Structure of the model

The model was developed in python using Keras and Tensorflow.

The input data was the patient data from 3 and 17 hours (in two different trial series) such that with only this data, the model should predict whether sepsis exists or not. Thus, inputting this data allows for the problem to be turned into one of binary classification.

While it may seem natural to use long short-term memory (LSTM) cells for a time-series data set, LSTM layers are not supported by Intel's neural compute stick 2 and thus the model was developed using dense layers. The model had 7 dense layers and a Leaky Rectified Linear Units (Leaky ReLU) activation function for the hidden layers. The output layer had a sigmoid activation function and binary cross entropy, as these are typically used with a binary classification problem. The nodes from for the 7 dense layers respectively were: 256, 128, 64, 64, 32, 32 and 1 node. A visualization of the model can be seen in **Figure 3** below.

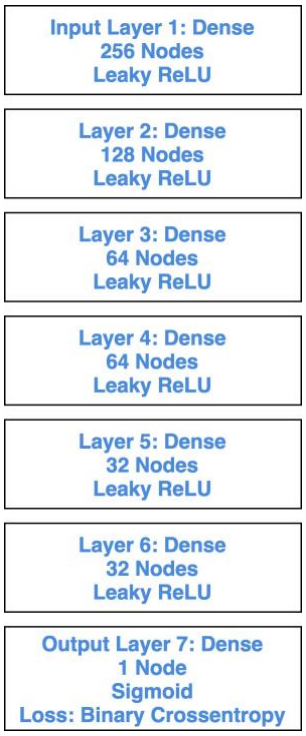

Figure 3: Visualization of the model

ii. Training of the model

The model was trained on approximately 36,000 patient data sets (50% sepsis, 50% no sepsis cases → from SMOT), with a validation set of 20,000 patient files (5% sepsis, 95% no sepsis, no SMOT). The model was trained up to 30 epochs and each the model was saved to find the best specificity and sensitivity of the model, which was calculated after each epoch. A noticeable trend was that as

sensitivity increased specificity decreased. An example of this can be seen in **Figure 4** below, as function of the number of epochs.

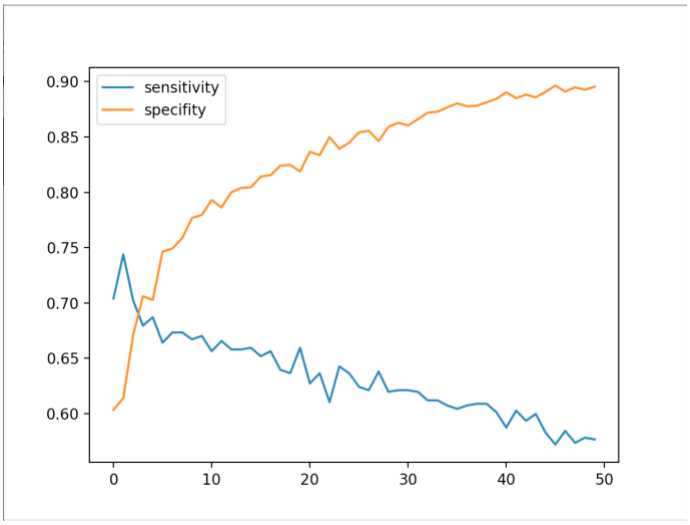

Figure 4: Specificity and Sensitivity vs. Number of Epochs. A noticeable trend is the increase in sensitivity results in a decrease of specificity.

While sensitivity to the disease is most important, specificity must also be retained such that only one class is not predicted. Thus, sensitivity and specificity must both be above 70%.

4. Implementation on Intel's Movidius Myriad X VPU and CPU

After the model was developed in Keras and trained on a CPU, it was then converted to Tensorflow framework as this is supported by Intel's OpenVINO Toolkit. This was done on Windows 10 Software on an x86-64bit processor. This was then converted to an intermediate representation (IR) file which was then implemented on Intel's Movidius Myriad X VPU (Neural Compute Stick 2) and a CPU.

5. Results

a. The Model

Given the model described above, using a validation data set of 20,000 patient files (5% of which were sepsis cases), 3 and 17 hours of patient data and all 40 parameters (Note: using the other sets of parameters were less successful yielding less than 70% sensitivity and specificity), the sensitivity (true positive rate) and specificity (true negative rate) of the model were 76% and 80% respectively. This indicates that the model successfully detected 76% of cases of sepsis and successfully classified 80% of healthy patients. Using only 3hrs of patient data the sensitivity (true positive rate) and specificity (true negative rate) of the model on the validation data were 72% and 71% respectively.

b. Implementation on Intel's Myriad X VPU

The model was successfully implemented on Intel's Neural Compute stick 2 (Myriad X VPU). The average time per patient file to predict an outcome was 10ms (predicting approximately 20,000 patients

within 3 minutes). This is a very fast timeframe for predication and seems feasible for bedside patient diagnostics/early detection. Additionally, Intel's Neural Compute Stick 2 costs $80 and is a USB device, reflecting both low cost and portability of the device increasing its clinical feasibility.

6. DISCUSSION

Overall, the results of the model predict sepsis cases well with 3hrs (72% sensitivity and 71% specificity) and 17hrs (76% sensitivity 80% specificity) of patient data. While these results are reasonable for an initial model, results yielding above 80-90% specificity and sensitivity would be more reasonable for clinical practice. The implementation of the model on the neural compute stick 2 was also successful with low patient prediction times from the deep learning model.

7. NEXT STEPS AND FUTURE WORK

This project has many potential next steps and future work. The three main areas for future work and research include, data preprocessing, the model and implementation of the neural network. Within data-preprocessing, there are many areas of research, including how to best deal with missing values. Papers have suggested different methods including masking missing values, zero-padding and many more methods. Given the volume of missing data, optimizing these values could give the model a significantly higher accuracy. Additional data preprocessing would include further optimizing the parameters used to improve the accuracy of the model. The current results utilize all 40 parameters of the model. While the other two selected parameter sets may not have yielded good results, there is likely further parameter optimization. Given that these results used all the parameters, optimizing which parameters best predict sepsis could not only increase the accuracy of the model, but clinically allow for more important parameters to the taken/focused on.

Within the model itself, current work being done includes optimizing the hyperparameters of the model to maximize specificity and sensitivity of the model including number of nodes, activation functions etc. Future work on the model includes making the model using LSTM cells in order to utilize the time series dataset, as LSTM cells are often more conducive to time-series data. The main focus in the next steps is tuning the hyperparameters of the model.

8. CONCLUSION

Deep learning within healthcare is an emerging and promising field to improve disease detection and patient outcomes. Successful deep learning models and implementation of deep learning algorithms could allow for more accurate, faster and earlier diagnosis of patients. This could lead to more treatment options for patients which could ultimately lead to a decreased mortality rate for many diseases. In particular, the applications of early diagnosis are particularly pertinent to time sensitive diseases including sepsis, which this project focused on. The deep learning model and data preprocessing allowed the model to have 72% sensitivity and 71% specificity with only 3 hours of patient data and 76% sensitivity and 80% specificity for 17 hours of patient data. While this specificity and sensitivity are reasonable, for clinical application or assisted diagnosis, sensitivity and specificity above 80-90% would be preferred. Hence future work and next steps includes more data preprocessing and hyperparameter tuning to improve the specificity and sensitivity of the model. Additionally, the successful implementation of the model on Intel's neural compute stick 2 indicates clinical feasibility through its speed and portability. This project found that the neural compute stick 2 required an average of 10ms to diagnose each patient given the patient file, proving speed, while the small USB nature, suggests portability. Although the model requires improvements to increase its accuracy, the model produced from this project predicts sepsis well and underscores the clinical viability of implementing machine learning models for assisted diagnosis in hospitals.

9. REFERENCES

Abramoff M.D., Lavin P.T., Birch M., Shah N., & Folk J.C. (2018). Pivotal trial of an autonomous AI-based diagnostic system for detection of diabetic retinopathy in primary care offices. *Yearbook of Paediatric Endocrinology*, *39*. doi: 10.1530/ey.16.12.1

Centers for Disease Control and Prevention. (2016, August 23). Data & Reports. Retrieved 2019, from https://www.cdc.gov/sepsis/datareports/index.html.

Ding Y., Sohn J.H. , Kawczynski M.G., Trivedi H., Harnish R., Jenkins N.W., Dmytro Lituiev, A... (2019). Deep Learning Model to Predict a Diagnosis of Alzheimer Disease by Using 18F-FDG PET of the Brain. *Radiology, 290*(2), 456-464. doi: 10.1148/radiol.2018180958.

Haenssle H A, Fink C, Schneiderbauer R, Toberer F, Buhl T, Blum A … Enk A, Uhlmann L, Reader study level-I and level-II Groups, Man against machine: diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 dermatologists, *Annals of Oncology*, Volume 29, Issue 8, August 2018, Pages 1836–1842, https://doi.org/10.1093/annonc/mdy166

Kumar, A., Roberts, D., Wood, K. E., Light, B., Parillo, J. E., Sharma, S., & Suppes, R. (2006). Duration of hypotension before initiation of effective antimicrobial therapy is the critical determinant of survival in human septic shock. *Crit Care Med*, *34*(6), 1589–1596. doi: 10.1097/01.CCM.0000217961.75225.E9

Paoli, C. J., Reynolds, M. A., Sinha, M., Gitlin, M., & Crouser, E. (2018). Epidemiology and Costs of Sepsis in the United States—An Analysis Based on Timing of Diagnosis and Severity Level*. *Critical Care Medicine*, *46*(12), 1889–1897. doi: 10.1097/ccm.0000000000003342

Reyna, M., Josef, C., Jeter, R., Shashikumar, S., Moody, B., Sharma, A., … Clifford, G. (2019, August 5). Early Prediction of Sepsis from Clinical Data -- the PhysioNet Computing in Cardiology Challenge 2019. Retrieved June 2019, from https://physionet.org/content/challenge-2019/1.0.0/.

Safaeian, M., & Solomon, D. (2009). Cervical Cancer Prevention - Cervical Screening: Science in Evolution. *Obstet Gynecol Clin North Am*, *34*(4), 739-ix. doi: 10.1016/j.ogc.2007.09.004

Seymour, C. W., Gesten, F., Prescott, H. C., Friedrich, M. E., Iwashyna, T. J., Philips, G. S., … Terry, K. M. (2017). Time to Treatment and Mortality during Mandated Emergency Care for Sepsis. *N Engl J Med*, *376*(23), 2235–2244. doi: 10.1056/NEJMoa1703058.

Shimabukuro, D. W., Barton, C. W., Feldman, M. D., Mataraso, S. J., & Das, R. (2017). Effect of a machine learning-based severe sepsis prediction algorithm on patient survival and hospital length of stay: a randomised clinical trial. *BMJ Open Respiratory Research*, 4(1). doi: 10.1136/bmjresp-2017-000234

Singer M, Deutschman CS, Seymour CW, et al. The Third International Consensus Definitions for Sepsis and Septic Shock (Sepsis-3). *JAMA.* 2016;315(8):801–810. doi:10.1001/jama.2016.0287

Soffer, S., Ben-Cohen, A., Shimon, O., Amitai, M. M., Greenspan, H., & Klang, E. (2019). Convolutional Neural Networks for Radiologic Images: A Radiologist's Guide. *Radiology*, *290*(3), 590–606. doi: 10.1148/radiol.2018180547.

World Health Organization. (2018, April 19). Sepsis. Retrieved from https://www.who.int/news-room/fact-sheets/detail/sepsis.