12-2021

Predicting risk of readmission in heart failure patients using electronic health records

Pradumna Suryawanshi
ps4558@rit.edu

Predicting risk of readmission in heart failure patients using electronic health records

APPROVED BY

SUPERVISING COMMITTEE:

Dr. Linwei Wang, Supervisor

Dr. Christopher Homan, Reader

Dr. Rui Li, Observer

Predicting risk of readmission in heart failure patients using electronic health records

by

Pradumna Suryawanshi

THESIS

Presented to the Faculty of the Department of Computer Science
Golisano College of Computer and Information Sciences
Rochester Institute of Technology

in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

Rochester Institute of Technology
December 2021

Acknowledgments

I wish to thank my supervisor for guiding me through the entire thesis and helping me understand intricacies of papers and how to analyze a paper. I would also like to thank Ryan for his valuable inputs while discussing results and papers. Lastly I would like to thank the Geisinger group for providing the dataset and clarifying multitudes of doubts.

Abstract

Predicting risk of readmission in heart failure patients using electronic health records

Pradumna Suryawanshi, M.S.
Rochester Institute of Technology, 2021

Supervisor: Dr. Linwei Wang

This thesis research investigates the prediction of readmission risk in heart failure patients using their electronic health record (EHR) data from previous hospitalizations. We examine three primary questions. First, we study the use of attention mechanism in readmission prediction model based on long short-term memory(LSTM) networks, and investigate the interpretability it offers regarding the importance of critical time during the visit in readmission prediction. Second given that, generally dataset is curated by combining data from multiple hospitals we investigate model generalization across multiple sites. Finally since in real life scenario model will be trained on past data and used to predict future readmission events ,we further investigate model generalization across time.Along with those things, model performance across different endpoints will be studied.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.0.1 Motivation

Heart failure has a considerable prevalence in western countries and accounts for approx 1-5 % of all hospitalization and approx 30 % of avoidable admissions[29]. Heart failure affects approx 6.5 million Americans with over 960K new cases each year. It results in approx $30.7 billion in hospitalization and associated costs. Billions of dollars can be saved by detecting avoidable admissions. Reducing avoidable admissions can also help to reduce stress on hospitals and other related services and resources. Recently due to the use of EHR(Electronic Health Data) systems the patient's records can be digitally saved and used for analysis later on. This has led to the accumulation of patient data which can be used to analyze and gain insights.

Recently neural networks have been use to solve complex problems like speech recognition[30], image recognition [16], object detection[13] etc. Availability and development of the GPU in last decade have helped the growth of neural networks. Large neural networks are able to achieve near human level performance on image recognition tasks. Leveraging the recent developments in deep learning various methods have come up to use deep learning to predict 30-day readmission .

Various datasets [20, 24] have been curated for the same, with minor subtleties among them. Many methods have been developed to deal with EHR data and prediction related to heart failure.Various methods of using LSTM for EHR data modeling and predicting readmission, length of stay, mortality have come up in recent times[1, 7, 40].They have used different subsections of EHR data, some have used values like blood pressure, lab values (Sodium, glucose, potassium, calcium etc) measured during the stay, some have used diagnosis at the start and end of the visit, some have used clinical notes for the task. Different aspects of EHR data have been studied like imputation of missing data[34],time difference between measured data points [32] irregular nature of time series and its modeling [33] etc.

This study mainly focuses on three things, first the use of attention mechanism for heart failure readmission prediction and identifying important time steps in visits, second, EHR datasets are curated by combining data from multiple hospital sites,we investigate the generalization across hospital sites.At last, we look into how data from the past affect the model and investigate model generalization across time.

1.0.2 Objective

The main objective of the study is to predict the risk of heart failure readmission patients. The study will analyze the effect of attention on hospital readmission risk and identify the important time steps in visit, it will also look at model generalization across various hospital sites and the effect of data generalization on model across time.It will also investigate the effect of changes in endpoint definitions on model performance.

Chapter 2

Background

Here we will look at general terms and definitions.

2.1 2-Approaches

Modeling irregular time series can be done with 2 different approaches.

2.1.1 Discrete

In this type of modeling the data is aggregated in a bin (in this case hours) such that all measurements of the variable within that bin is passed through an aggregation function to get a single value. The function used is mean/max. The experiments use discrete time series modeling.

2.1.2 Continuous

In this type of modeling the data(irregular time series) is not aggregated in bins of some time unit , instead available data points are used to generate a continuous representation and feed to model [21].

2.2 Neural Networks and its Applications in Medical Domain

Neural networks have been used to solve real life problems like image recognition and object detection[14].Various complex networks have been developed over the last two decades for specific applications, like CNN for image recognition , object detection etc. Along with the development in processing power and availability of large datasets [9] have helped neural networks achieve human level performance. Neural networks have been used to solve medical problems for the past 2 decades. [12]. CNN (Convolutional neural network) which use the spatial data from the image to analyze the image and gain information about objects and predict some useful information has been used to detect tumors [3] from MRI and analyze scans for possible anomalies [23].

RNN(Recurrent neural networks) are a type of neural network that deals with input that has a time component, for example, output from a sensor that measures temperature [15] , humidity [11] , heartbeat of a person in hospital etc, stock price etc [27].

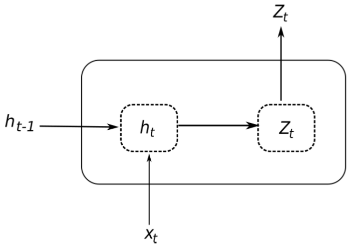


Figure 2.1: ERNN cell
[17]

2.3 Recurrent Neural Network

RNN has proved to be successful in modeling time series inputs over traditional statistical models. It's architecture marks a important change in the way neural networks analyze time series data, by incorporating the temporal aspect of data. RNN was introduced in 1990 by Elman [10] , architecture is as shown in figure 2.1.

$$\begin{gathered} h_t = \sigma\left(W_i \cdot h_{t-1} + V_i \cdot x_t + b_i\right) \\ z_t = \tanh\left(W_o \cdot h_t + b_o\right) \end{gathered} \tag{2.1}$$

In the above equations which defines the operation of RNN $h_t \in R^d$ is defined as the d - dimensional hidden state , which is updated every time step. $x_t \in R^m$ is a m-dimensional input feed at each time-step to the cell.$W_i \in \mathbb{R}^{d \times d}$ along with $V_i \in \mathbb{R}^{d \times d}$is the input side weight matrix which is optimized as a part of training process, A optional d-dimensional bias matrix $b_i \in R^d$ is also present. Like wise $W_o \in \mathbb{R}^{d \times d}$ and $b_o \in R^d$ are output side weight matrix and bias matrix respectively. The cell uses two activation functions sigmoid σ and tanh which help to convert the linear operations to non-linear operations. The current hidden state is updated using the previous hidden state and current input. These computations help the cell learn from past and better understand the current input. The input can be arbitrarily long , as long as all examples in training set are of same length, which can be trivially achieved using padding. The basic cell suffers from vanishing gradients problems as the sequence gets long , keeping track of gradient at earlier stages of sequence becomes difficult. When the gradients are large , the gradients tend to explode causing exploding gradient problem which limits the performance of the cell.[31] Various versions of the cell have been proposed over time to overcome the drawback of vanishing gradients like LSTM(Long Short Term memory) [18], GRU(Gated Recurrent Unit) [6].

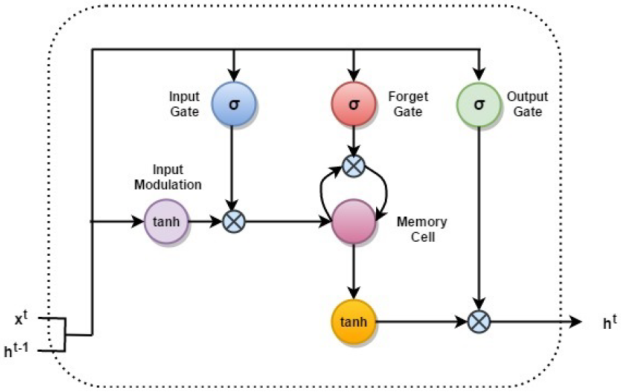


Figure 2.2: LSTM Cell
[5]

2.4 LSTM-Long Short Term Memory

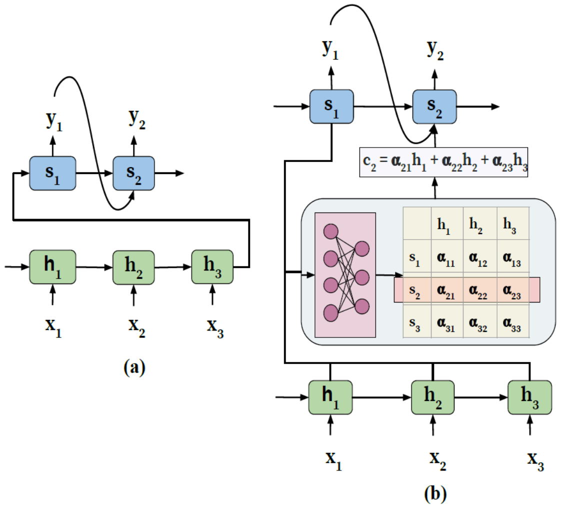
LSTM was introduced in 1997 by Hochreiter and Schmidhuber [18].LSTM has achieved better performance modeling time series data like music generation, speech signal analysis [8]. It has addressed the vanishing gradient issue in ERNN to a great extend. It has a extra cell state along with the hidden state of the ERNN cell.

$$\begin{gathered} i^t = \sigma\left(W_i x^t + U_i h^{t-1} + b_i\right) \\ f^t = \sigma\left(W_f x^t + U_f h^{t-1} + b_f\right) \\ o^t = \sigma\left(W_o x^t + U_o h^{t-1} + b_o\right) \\ a^t = \tanh\left(W_c x^t + U_c h^{t-1} + b_c\right) \\ c^t = i^t \cdot a^t + f^t \cdot c^{t-1} \\ h^t = o^t \cdot \tanh\left(c^t\right) \end{gathered} \tag{2.2}$$

The equations above define the structure of LSTM cell which is used for modeling time series data like language [28] ,stocks [26], music [25], speech [39] etc. The equations represents computation for time step t. i_t, f_t, o_t represent the input ,forget and output gates of the cell. $W_i, U_i \in R^{dxd}$ are matrix of input gate and input respectively. $W_o, U_o \in R^{dxd}$ are matrix of output gate and input respectively. $W_f, U_f \in R^{dxd}$ are matrix of forget gate and input respectively. $x^t \in R^d$ is the d-dimensional input at time step *t*. h^t is d-dimensional hidden state of the cell which is updated with every time step *t*. c^t is the current cell state and a^t is the candidate cell that decides how much information should be propagated to future. c^{t-1} is previous cell state of the cell at time step *t-1*. h^t is the current hidden state of the cell at time step *t*. h^{t-1} is previous hidden state of the cell at time step *t-1*. h^t along with hidden state it can also be used as the output of cell at time step *t*.

2.5 Attention Mechanism

Attention was introduced to overcome the drawbacks of encoder-decoder architecture for machine translation [2]. Attention mechanism in neural network has been inspired by how human brain processes image system, when identifying a person in a room the human brain focuses it's attention on facial aspects of the person rather than the flower pot behind him, therefore focusing a particular part of input will help gain sufficient insights from input[36]. On similar lines image have some parts that are more important than other [35], in language a set of words are more important than others for example in translation [2] a part of sentence has more insights than rest of the sentence. Attention mechanism helps the model to find such important parts dynamically and making it pay more attention to such part which in turn helps the model perform better by paying less attention to other parts. Few of the main drawbacks of encoder -decoder architecture are, it converts the entire input sequence to a single vector at the end essentially compressing the entire sequence to a single vector, which results in loss of information[6], it also suffers from the alignment modeling between input and output sequence[38]. As the figure 2.3 shows basic difference between a traditional encoder and decoder is the context vector that will add weight to input before feeding it to decoder. The context vector helps introduces attention to the input.The weights are learnable parameter and can be optimized for desired task.



	h_1	h_2	h_3
s_1	α_{11}	α_{12}	α_{13}
s_2	α_{21}	α_{22}	α_{23}
s_3	α_{31}	α_{32}	α_{33}

Figure 2.3: (a) traditional encoder-decoder, (b) encoder decoder with attention [4]

2.5.1 Hierarchical Attention

This type of attention was first introduced in [37] Jan 2016, it follows a completely different approach to attention, which does not use encoder-decoder model. It was developed for classifying documents and has a unique hierarchical structure that mirrors the structure of document i.e words form a sentence , sentences form a document. It uses a cascading RNN layer where each sentence is passed through a RNN and then a vector representing the sentence is generated using attention mechanism on top of RNN which then is passed through second RNN as shown in the figure 2.4. The second RNN takes as input the transformed sentences as input and at outputs the classification score for the document.

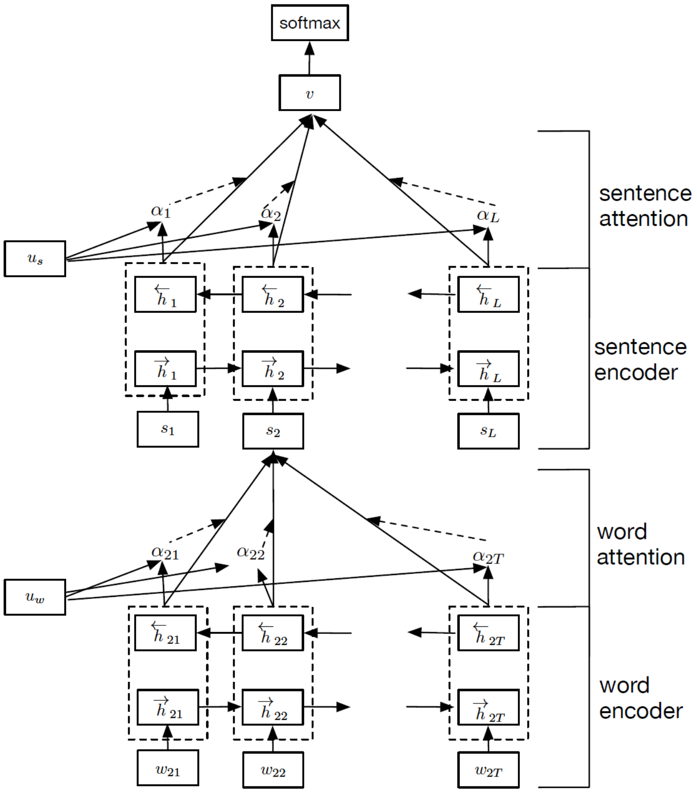


Figure 2.4: Hierarchical Attention Model
[37]

$$\begin{aligned} x_{it} &= W_e w_{it}, t \in [1, T] \\ \overrightarrow{h}_{it} &= \overrightarrow{\text{RNN}}\left(x_{it}\right), t \in [1, T], \\ \overleftarrow{h}_{it} &= \overleftarrow{\text{RNN}}\left(x_{it}\right), t \in [T, 1] \end{aligned} \tag{2.3}$$

The above equation defines the operations associated with first level of RNN's. Given a sentence with words w_{it} , words are converted to vector using embedding matrix W_e. A sentence contains T words, each sentence is passed through a RNN to get compressed weighted representation of the sentence.Each word is assigned a specific weight which generating the final compression representation for that sentence.

$$\begin{aligned} u_{it} &= \tanh\left(W_w h_{it} + b_w\right) \\ \alpha_{it} &= \frac{\exp\left(u_{it}^{\top} u_w\right)}{\sum_t \exp\left(u_{it}^{\top} u_w\right)} \\ s_i &= \sum_t \alpha_{it} h_{it} \end{aligned} \tag{2.4}$$

The above equation defines word level attention after first layer of RNN. The word level vector i.e combination of $h_{it} = \overrightarrow{h}_{it} + \overleftarrow{h}_{it}$ (forward and backward hidden states) is then passed through a single layer neural network to get u_{it} which is the latent representation of h_{it}. The importance of word is generated by measuring the similarity u_{it} with u_w as word level context vector and normalized score a_{it} is generated after applying softmax function. At the end we compute the representation of sentence which is then passed to next RNN level as S. A similar approach is used for sentence level attention at the end of second RNN.

A similar approach can used in patient care model which tries to predict 30 day readmission by identifying certain hours using attention which are more important than the other and helping the model to pay more attention (weighted values) to those hours . The proposed modification of hierarchical attention mechanism uses only word level attention to identify important hours.

Chapter 3

Methodology

3.1 Dataset

The dataset consists of EHR data from the Geisinger group and consists of $\sim$15K unique patients and $\sim$30K visits over a span of $\sim$20 years. The study will use the first 48 hours of data of a patient's visit to predict if the patient will have readmission in the next 30 days.The data has been aggregated to generate 12 hr time steps from 48-hour raw data. In this case we let $\mathcal{D} = \{(x_n, y_n)\}$ where y_n is single dimensional target $\epsilon = \{0, 1\}$ where y_n takes on 1 if the patient will have a readmission in next 30 days of discharge.Readmission is defined as case where patient is admitted again within 30-days of discharge from hospital. x_n is a q dimensional combination of time series data and static data. Time series variables used are as follows :-

- BP Diastolic
- BP Systolic
- MEWS Score
- Glasgow-Best Motor Response
- Glasgow-Best Verbal Response

- Glasgow-Eyes Open
- Glasgow-Total Score
- Pulse
- Urine Output
- $Fi0_2$
- Mean Arterial Pressure
- Resp
- SpO_2
- Temp

Where a single time series is represented as $x_{nq} = [x_{qn1}, x_{qn2}, ...]$. Additional derived features extracted from time series variables such as min, max , mean over measured interval are added as static values. Static data variables such as demographics which includes gender, age and insurance type are also added to the time series data.Derived features from lab results such as min, max, mean added as static variables at end of every time step.Derived features from patients history such as

- Number of encounters last year.
- Number of encounters in last 6 months.

- Number of days hospitalized last year.
- Number of days hospitalized in last 6 months.
- Number of emergency visits last year.
- Number of emergency visits exceeding 5 days last year.

are also added as a static feature along every time step These features help track the progression of disease.Linear regression coefficients of individual time series and lab values also added as static features.Lab values used are :-

- Glucose Meter
- Potassium
- Creatine
- Sodium
- Chloride
- CO2
- Calcium
- HGB
- HCT

Min-Max normalization is used to make sure the data is scaled appropriately. Missing data is imputed using mean of the cohort.If a variable has multiple measurement in a hour the values are aggregated using mean function. A train test split of 80 %(80% train -20% validation)-20% is used. Varied number of visits have been registered form different hospitals in the dataset(for dataset version 1).

Table 3.1: Distribution of visits across hopsitals

Hospital Name	**Number of visits**
GMC	13713
GBH	515
GLH	2336
GSACH	1688
GWV	8087
GCMC	3991
Total	30340

As seen from the table the largest hospital is GMC and the smallest being GBH with respect to number of visits from each hospital in the dataset. The model consists of a 2 layer deep LSTM with 4 hidden layers on top along with the additional attention mechanism and hidden size of 1024.The optimizer used is SGD [22] with a learning rate of lr=0.001.

3.2 Procedure

The data is curated from a global pool of visits from Geisinger data. First, the patients with HF diagnosis at least once during their lifetime are selected. All visits from such patients which span at least 48 hours are selected.This forms the first set of visits. The second set consists of patients that have heart failure diagnosis with at least 48 hours of data and a 30-day readmission diagnosis of heart failure. The third set consists of patients who have been diagnosed with heart failure at least once in their lifetime and have all-cause readmission within 30 days of discharge [19]. The 3 different sets of data define the definitions of the endpoints used in literature.

Feature extraction takes place using LSTM and a fully connected layer is used to extract from the hidden states of the LSTM. A modified version of hierarchical attention is used on top of LSTM to identify the important hours of visit.

3.3 Experiments

Here we will look at the experiments conducted.

3.3.1 Experiment 1

This experiment deals with identifying a better model, LSTM or LSTM with attention. For this we use the first set of visits as described in previous section. A test-train-validation split is randomly generated and is used to compare both models. Both models i.e LSTM and LSTM+ attention are trained using the same train and validation split and tested on same test split.

3.3.2 Experiment 2

This experiment deals with comparing performance of model trained composite hospital data vs model trained on individual hospital data.First set of visits as described above are used for the experiment.Both models are trained and tested on same test-train-validation split respectively.

3.3.3 Experiment 3

This experiment tries to identify the effect of old data on the model and its performance. The earliest visits date from 2002 and the latest dates from 2020. In this experiment the test-train-validation splits are segregated by year i.e data from 2010-2017 is used for training and validation, data from 2017-2020 is used for testing.Then the training data is increased by one year and a new model is trained, based on the performance of model,effect of recent vs old data is analyzed.Looking back 10 years from 2014 with increments of one year the training set is generated and corresponding data after 2014 is used for testing. The setting of this experiment represents a real life scenario in hospitals where a patient might have old visits from recent years in the database and prediction on his current visit is performed.

3.3.4 Experiment 4

This experiment analyzes the effect of endpoint definition. It has 3 different experiments listed below :

- Using only heart failure specific visits approx 8K vists.

- Using all visits which are related to heart problems from patients who have diagnosed with heart failure at least once in lifetime , approx 31 K visits
- Using all cause visits from patients who have diagnosed with heart failure at least once in lifetime approx 46K visits.

These 3 are the most popular endpoints definitions, changing the definitions affects dataset as well as readmission rate which in turn affects the models performance.The experiment will analyze the effect of these endpoints on model performance.

Chapter 4

Results and Discussion

4.1 Experiment 1

Here we will look at results for LSTM vs LSTM+ attention model.The model is trained on dataset version 1 as described before(all data split into train-validation-test).

Table 4.1: Results for LSTM vs LSTM+attention

Model Name	**Score-ROC**
LSTM	0.61
LSTM+Attention	0.66

As we can see from table 4.1 the model with attention has performed better than model with LSTM, A increase of approx 0.05 points or increase of approx 8% has been observed.Attention basically helps the model to look at time steps which are important and assign more importance to them while predicting.

4.2 Experiment 2

In this experiment, the performance of models trained for individual hospital vs model trained for all hospitals is analyzed. The composite in tables below represents model trained on all data, and tested only for the particular hospital. The test splits are same for composite and individual model.

Table 4.2: Results for GMC hospital

Model Name	**Score-ROC**
Compsite	0.57
GMC	0.65

As seen in the table, model trained on individual(GMC) hospitals data has performed better as compared to model trained on data from all hospitals combined, the model was able to gain better insights without the noise from other hospitals. Improvement of approx 0.08 points or a increase of 14% has been seen.

Table 4.3: Results for GSACH hospital

Model Name	**Score-ROC**
Compsite	0.58
GSACH	0.66

As seen in the table, model trained on individual hospitals data has performed better as compared to model trained on data from all hospitals combined, the model was able to gain better insights without the noise from other hospitals. Improvement of approx 0.08 roc points or a increase of 13% has been seen.

Table 4.4: Results for GWV hospital

Model Name	**Score-ROC**
Compsite	0.56
GWV	0.61

As seen in the table, model trained on individual hospitals data has performed better as compared to model trained on data from all hospitals combined, the model was able to gain better insights without the noise from other hospitals.Improvement of approx 0.06 points or a increase of 10% has been seen.

As seen in the table, model trained on individual hospitals data has performed better as compared to model trained on data from all hospitals combined, the model was able to gain better insights without the noise from

Table 4.5: Results for GCMC hospital

Model Name	**Score-ROC**
Compsite	0.52
GMC	0.61

other hospitals.Improvement of approx 0.09 points or a increase of 17% has been seen.

Table 4.6: Results for GLH hospital

Model Name	**Score-ROC**
Compsite	0.58
GLH	0.73

As seen in the table, a model trained on individual hospitals data has performed better as compared to a model trained on data from all hospitals combined, the model was able to gain better insights without the noise from other hospitals. Improvement of approx 0.15 points or an increase of 25% has been seen.

The models trained on individual hospitals have outperformed model trained on composite data by atleast 10% or atleast 0.08 roc points, which is a significant improvement. The hospitals have varied visits proportions in the composite dataset while GMC accounting for approx 45% of all visits vs GBH which accounts for only 1.6% , which is also the reason GBH was not used for analysis as training models on GBH was a difficult task.

4.3 Experiment 3

This experiment analyses the performance of time based data i.e using a couple of years old data to train model and analyze the effect of recent vs old data. The models are trained using 2 year old data to 8 year old data. Anchor year i.e year used to split train test year data, data(visits) after the anchor year is used as test set and data(visits) before anchor year is used as a train set. Multiple train sets are generated by changing the lookback window i.e a set is generated by looking back 2 years from anchor year, another set is generated by looking back 3 years in the past from the anchor year, and so on.

Table 4.7: Results (LB:look back period)

Anchor year	**LB-2**	**LB-3**	**LB-4**	**LB-5**	**LB-6**	**LB-7**	**LB-8**
2017	0.53	0.54	0.60	0.59	0.61	0.61	0.62
2016	0.51	0.52	0.59	0.60	0.61	0.62	0.62

As seen from table 4.7 we can see the recent data from last 4-5 years proves to be more effective as compared to old data. Data from recent past has more effect on model than old data.

4.4 Experiment 4

This experiment deals with the analyzes of how model performance varies with changes in end-point definitions. The experiment analyzes 3 different end points ,

- Using only heart failure specific visits.
- Using all visits which are related to heart problems from patients who have diagnosed with heart failure at least once in lifetime.
- Using all cause visits from patients who have diagnosed with heart failure at least once in lifetime.

Table 4.8: Results for endpoint changes

Parameter	**HF-Specific datatset**	**Heart Related**	**All cause**
Score	0.69	0.66	0.63
Dataset Size	8K	30K	47K
Readmission Rate	7%	13%	13%

The performance of the model fluctuates by around 0.04 points roc by changing the end-point definitions. The changes are also a result of readmission rates and datasets size fluctuations. Including all cause visits also increase the visits that are not related heart failure and related causes, which further induces noise in the dataset.The readmission rate also varies a lot with end-point changes which essentially increased by 50% for the heart related and all cause dataset when compared to hf-specific dataset.

Chapter 5

Conclusion

5.1 Conclusion

After completing these experiments we can conclude that using attention has its benefits and adds discrimination power to the model. Developing a single model for a hospital performs better than using a model trained with the composite dataset. A model trained on recent data for the last 4-5 years performs better as compared to a model trained on older data, the model does not improve a lot by adding older data. Changing the definitions of the endpoints i.e how readmission is defined affects the performance of the model, because it changes the readmission rate and size of datasets. By analyzing the results, a stricter readmission definition related to a specific problem helps the model perform better as compared to a loosely defined readmission definition.

Bibliography

[1] Merhan A. Abd-Elrazek, Ahmed A. Eltahawi, Mohamed H. Abd Elaziz, and Mohamed N. Abd-Elwhab. Predicting length of stay in hospitals intensive care unit using general admission features. *Ain Shams Engineering Journal*, 12(4):3691–3702, 2021.

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate, 2016.

[3] Md. Abu Bakr Siddique, Shadman Sakib, Mohammad Mahmudur Rahman Khan, Abyaz Kader Tanzeem, Madiha Chowdhury, and Nowrin Yasmin. Deep convolutional neural networks model-based brain tumor detection in brain mri images. *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, Oct 2020.

[4] Sneha Chaudhari, Varun Mithal, Gungor Polatkan, and Rohan Ramanath. An attentive survey of attention models, 2021.

[5] Thomas Cherian, Akshay Badola, and Vineet Padmanabhan. Multi-cell lstm based neural language model. *arXiv preprint arXiv:1811.06477*, 2018.

[6] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation, 2014.

[7] Edward Choi, Andy Schuetz, Walter F Stewart, and Jimeng Sun. Using recurrent neural network models for early detection of heart failure onset. *Journal of the American Medical Informatics Association*, 24(2):361–370, 08 2016.

[8] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[10] JL Elman. Finding structure in time. cog. *Sci*, 14:179–211, 1990.

[11] Chen Fang, Xipeng Wang, Yi L Murphey, David Weber, and Perry MacNeille. Specific humidity forecasting using recurrent neural network. In *2014 International Joint Conference on Neural Networks (IJCNN)*, pages 955–960, 2014.

[12] Jari J. Forsström and Kevin J. Dalton. Artificial neural networks for decision support in clinical medicine. *Annals of Medicine*, 27(5):509–517, 1995.

[13] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation, 2014.

[14] Ross B. Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *CoRR*, abs/1311.2524, 2013.

[15] Jung Min Han, Yu Qian Ang, Ali Malkawi, and Holly W. Samuelson. Using recurrent neural networks for localized weather prediction with combined use of public airport data and on-site measurements. *Building and Environment*, 192:107601, 2021.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[17] Hansika Hewamalage, Christoph Bergmeir, and Kasun Bandara. Recurrent neural networks for time series forecasting: Current status and future directions. *International Journal of Forecasting*, 37(1):388–427, Jan 2021.

[18] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[19] Mehdi Jamei, Aleksandr Nisnevich, Everett Wetchler, Sylvia Sudat, and Eric Liu. Predicting all-cause risk of 30-day hospital readmission using artificial neural networks. *PloS one*, 12(7):e0181173, 2017.

[20] Alistair Johnson, Tom Pollard, Lu Shen, Li-wei Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Celi, and Roger Mark. Mimic-iii, a freely accessible critical care database. *Scientific Data*, 3:160035, 05 2016.

[21] Patrick Kidger, James Morrill, James Foster, and Terry Lyons. Neural controlled differential equations for irregular time series, 2020.

[22] J. Kiefer and J. Wolfowitz. Stochastic estimation of the maximum of a regression function. *The Annals of Mathematical Statistics*, 23(3):462–466, 1952.

[23] Byungjai Kim, Kinam Kwon, Changheun Oh, and Hyunwook Park. Unsupervised anomaly detection in mr images using multi-contrast information, 2021.

[24] Shuyu Lu, Ruoyu Chen, Wei Wei, and Xinghua Lu. Understanding heart-failure patients ehr clinical features via shap interpretation of tree-based machine learning model predictions, 2021.

[25] Sanidhya Mangal, Rahul Modak, and Poorva Joshi. Lstm based music generation system. *arXiv preprint arXiv:1908.01080*, 2019.

[26] Sidra Mehtab and Jaydip Sen. Stock price prediction using cnn and lstm-based deep learning models. In *2020 International Conference on Decision Aid Sciences and Application (DASA)*, pages 447–453. IEEE, 2020.

[27] Adil Moghar and Mhamed Hamiche. Stock market prediction using lstm recurrent neural network. *Procedia Computer Science*, 170:1168–1173, 2020. The 11th International Conference on Ambient Systems, Networks and Technologies (ANT) / The 3rd International Conference on Emerging Data and Industry 4.0 (EDI40) / Affiliated Workshops.

[28] R. Monika, S. Deivalakshmi, and B. Janet. Sentiment analysis of us airlines tweets using lstm/rnn. In *2019 IEEE 9th International Conference on Advanced Computing (IACC)*, pages 92–95, 2019.

[29] Dariush Mozaffarian, Emelia J Benjamin, Alan S Go, Donna K Arnett, Michael J Blaha, Mary Cushman, Sarah De Ferranti, Jean-Pierre Després, Heather J Fullerton, Virginia J Howard, et al. Heart disease and stroke statistics—2015 update: a report from the american heart association. *circulation*, 131(4):e29–e322, 2015.

[30] Ali Bou Nassif, Ismail Shahin, Imtinan Attili, Mohammad Azzeh, and Khaled Shaalan. Speech recognition using deep neural networks: A systematic review. *IEEE Access*, 7:19143–19165, 2019.

[31] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty

of training recurrent neural networks. In *International conference on machine learning*, pages 1310–1318. PMLR, 2013.

[32] Safa Onur Sahin and Suleyman Serdar Kozat. Nonuniformly sampled data processing using lstm networks. *IEEE Transactions on Neural Networks and Learning Systems*, 30(5):1452–1461, 2019.

[33] Satya Narayan Shukla and Benjamin M. Marlin. Multi-time attention networks for irregularly sampled time series. *CoRR*, abs/2101.10318, 2021.

[34] Bhanu Pratap Singh, Iman Deznabi, Bharath Narasimhan, Bryon Kucharski, Rheeya Uppaal, Akhila Josyula, and Madalina Fiterau. Multi-resolution networks for flexible irregular time series modeling (multi-fit). *CoRR*, abs/1905.00125, 2019.

[35] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. Residual attention network for image classification, 2017.

[36] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention, 2016.

[37] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. pages 1480–1489, 01 2016.

[38] Tom Young, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria. Recent trends in deep learning based natural language processing, 2018.

[39] Albert Zeyer, Patrick Doetsch, Paul Voigtlaender, Ralf Schlüter, and Hermann Ney. A comprehensive study of deep bidirectional lstm rnns for acoustic modeling in speech recognition. In *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 2462–2466. IEEE, 2017.

[40] Yao Zhu, Xiaoliang Fan, Jinzhun Wu, Xiao Liu, Jia Shi, and Cheng Wang. Predicting icu mortality by supervised bidirectional lstm networks. In *AIH@ IJCAI*, 2018.

Vita

Pradumna Vilas Suryawanshi was born in Pota, India on November 08, 1995, the son of Vilas Suryawanshi and Jayshree Suryawanshi. He received the Bachelor of Engineering degree in Electronics and telecommunication from Pune Institute of Computer Technology, Pune, India in 2017. He is currently pursuing his Master of Science degree from Rochester Institute of Technology, United States of America. His research interest includes time series analysis, Computer Vision, Image Processing and Machine Learning. His current research includes time series analysis of patient EHR data for readmission prediction.

Permanent address: 709 parkpoint drive, unit 3
Rochester, New York 14623

This thesis was typeset with LaTeX[†] by the author.

[†] LaTeX is a document preparation system developed by Leslie Lamport as a special version of Donald Knuth's TeX Program.