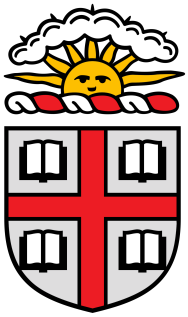
Predicting Hospital Readmission using Unstructured Clinical Note Data

Prakrit Baruah

Thesis Advisers: Carsten Eickhoff, George Zerveas
Thesis Reader: Ellie Pavlick



Department of Computer Science
Brown University
Providence, RI
May 2020

Contents

Abstract

Hospital readmission is costly for hospitals and is associated with worse outcomes for patients. Many readmissions are avoidable if patients receive further care during their initial admission. Therefore, a predictive model that can indicate whether a patient is likely to be readmitted is very valuable. A convolutional neural network architecture proposed by Zerveas *et al.* achieves state-of-the-art results in mortality prediction. This work explores the effectiveness of Zerveas *et al.*'s model in predicting hospital readmission. My results indicate that this model is much more effective in predicting hospital readmissions for the subset of patients with heart failure than for the overall dataset of patients with any disease. Furthermore, discharge summaries tend to improve the model's ability to predict readmission, especially in shorter-term time frames.

1 Introduction

Hospital readmission is both extremely costly and is associated with worse patient outcomes [11]. Although some readmissions are inevitable, many can be avoided with special attention during a patient's visit [11]. A median 27.1% of readmissions are avoidable, but estimates range from 5% to as high as 79% [15]. Even fractionally reducing patient readmission could save Medicare billions of dollars [11]. Therefore, determining the likelihood of hospital readmission is critical information for both patients and hospitals. Ideally, this may be determined by analysing information during a patient's initial admission.

Patient information is stored in electronic health records (EHRs). EHRs contain both structured data – categorical or numerical data which is organised, such a patient's blood pressure – and unstructured data – data which is not organised, such as doctors and nurses notes. Given that the data within an EHR paints an accurate and comprehensive picture of a patient, it can be used to predict hospital readmission. Machine learning is particularly well suited to this task.

Extensive work has already been conducted in analysing the structured data within an EHR to predict whether a patient is likely to be readmitted. Structured data is easier to deal with – numerical or categorical data can be easily fed into machine learning models. Structured data tasks leverage physiological factors, such as blood pressure and cholesterol, or non-physiological factors, such as race and gender. Futoma *et al.* utilise both linear models such as support vector machines and non-linear models such as deep learning techniques on structured data [4].

On the other hand, unstructured data poses a greater challenge. Clinical notes contain free text which reveal a much richer profile of a patient than structured data. A patient is likely to have dozens of clinical notes covering the duration of their admission. However, the free text must be pre-processed before it can be passed through a machine learning algorithm.

Most research utilising clinical notes uses a Bag-of-Words model to represent text data [17]. However, clinical note data is complex and the spatial relationship between words is often important. The Bag-of-Words model is therefore likely to oversimplify clinical note data. Word embeddings have been used to create more complex representations of clinical note data with great success [12]. Machine learning models, such as support vector machines, perform well with a Bag-of-Words or word2vec models [2]. More complex deep learning methods, such as recurrent neural networks, long-term short-term memory neural networks, and convolutional neural networks, have generally been more successful [10, 2, 13]. Unlike simpler machine learning models, these deep learning techniques can capture the temporal ordering of clinical notes. This allows the model to track the progression of a patient's admission which can therefore improve the model's predictive performance.

Zerveas *et al.* achieve state-of-the-art results in predicting patient mortality using convolutional neural network (CNN) based models on unstructured clinical note data [16]. Readmission prediction is a very similar task to mortality

prediction and, therefore, Zerveas *et al.*'s model should perform similarly well in the readmission task. I utilise this neural network architecture to tackle the problem of readmission prediction using clinical note text data.

1.1 Clinical Relevance

A model which accurately predicts hospital readmission would be very valuable. During a patient's admission, their likelihood of readmission would be continuously calculated given the data collected in the EHR. If readmission is deemed likely, greater resources and care could be diverted to improve the patient's outcome. Furthermore, since the average hospital readmission costs between $7,000 and $19,000, accurately predicting readmission would save hospitals and patients substantial resources [9].

2 Method

2.1 Data Exploration

I use the freely accessible MIMIC-III database for this task [7]. This data was collected from the critical care units of Beth Israel Deaconess Medical Center from 2001 to 2012. Although the database contains a wide range of information, I primarily use the patient admission, notes, and diagnosis data.

2.1.1 Diagnoses

The diagnosis information for patients is contained in the `diagnosis_icd` table [7]. There are 651,047 diagnoses for 46,520 patients. These diagnoses are recorded in the form of ICD9 codes. This is a standard classification system assigning codes to diagnoses and procedures. ICD9 codes are hierarchical in nature – a specific diagnosis falls under a parent group of diagnoses. Figure 1 shows an example of this hierarchy for the specific disease "cholera due to vibrio cholerae" which has the code "001.0."

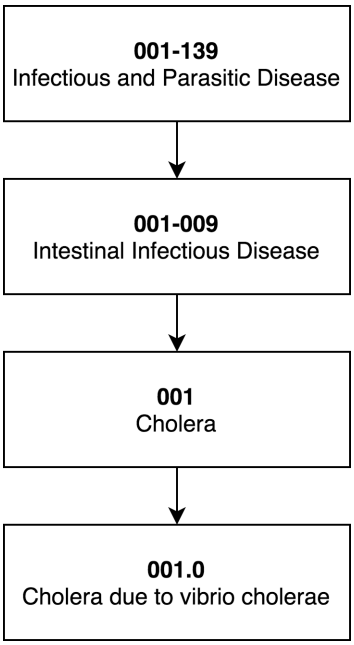


Figure 1: ICD9 code hierarchy for the disease "cholera due to vibrio cholerae"

Each patient has, on average, 14 diagnoses over all of their admissions. On average, each admission contains 11 diagnoses. Note that although a patient may have multiple diagnoses per admission, these diagnoses are priority-ordered by the `SEQ_NUM`. The primary diagnosis of a patient admission is therefore the

diagnosis with the lowest `SEQ_NUM` for that admission. Considering only the primary diagnoses for each admission, there are still 2,944 distinct ICD9 codes showing all primary diagnoses. Table 1 shows the frequency of the most common primary diagnoses.

Table 1: Frequency of the most common diagnoses

ICD9 Code	Count	Description
V30	6265	Single live born
414.01	3460	Of native coronary artery
038.9	2028	Unspecified septicemia
410.7	1721	Subendocardial infarction
424.1	1118	Aortic valve disorders
518.81	1111	Acute respiratory failure
431	1019	Intracerebral hemorrhage
V31	989	Twin, mate live born
486	710	Pneumonia, organism unspecified

Since there are over 6,985 different ICD9 codes in the `diagnosis_icd` table, I roll-up each ICD9 code to the most general parent diagnosis possible. For example, for the disease "cholera due to vibrio cholerae" in Figure 1, I take the parent diagnosis to be "infectious and parasitic disease." Note that the greatest frequency of a diagnosis in Table 1 is only 6,300 patients out of more than 46,000 patients. It is therefore helpful to generalise the diagnosis to detect any trend in the accuracy or precision of readmission prediction given a general diagnosis. Taking all possible ICD9 diagnoses would be too granular and therefore difficult to determine any trend in the predictive ability of the model for a given group of diseases.

Table 2: Frequency of the most common diagnoses after "rolling up" specific diseases to parent diseases. N&M indicates nutritional and metabolic diseases.

Parent ICD9 Code	Count	Description
390-459	18224	Diseases of the circulatory system
900-999	8288	Injury and poisoning
V30-V39	7691	Live born infants according to type of birth
520-579	5167	Diseases of the digestive system
460-519	4700	Diseases of the respiratory system
001-139	4134	Infectious and parasitic diseases
140-239	3723	Neoplasms
240-279	1559	Endocrine, N&M diseases, and immunity disorders
580-629	1080	Diseases of the genitourinary system
320-389	848	Diseases of the nervous system and sense organs

Table 2 shows the frequencies of general diagnoses for each patient. The most

common group of diagnoses concerns the circulatory system. Other researchers have trained models to predict hospital readmission amongst patients who have congestive heart failure [10]. Readmission labels for this particular sub-group of patients are less class-imbalanced because patients with heart failure have much higher readmission rates than patients with other diseases [1]. This work therefore uses not only patients with any disease as a target, but also the subset of patients with heart failure as a target to evaluate the proposed model. To do this, I create a heart failure dataset by filtering the general dataset on specific ICD9 codes. I use the same ICD9 codes as Liu *et al.* to determine patients with congestive heart failure: 398.91, 402.01, 402.11, 402.91, 404.01, 404.03, 404.11, 404.13, 404.91, 404.93, 428.0, 428.1, 428.20, 428.21, 428.22, 428.23, 428.30, 428.31, 428.32, 428.33, 428.40, 428.41, 428.42, 428.43, and 428.9 [10].

The third most frequent group of diagnoses concerns newborns. However, according to Huang *et al.*, newborns have a very particular distribution of clinical notes and readmission labels [5]. This can make it difficult for a machine learning model to generalise trends which indicate readmission. They are therefore excluded from the overall dataset.

2.1.2 Admissions

The MIMIC-III dataset contains information on 58,976 unique admissions for 46,520 patients. The majority of patients are only admitted to the hospital once. Table 3 shows the frequency of the number of admissions per patient. Most patients are admitted only once, and around 11% of patients are admitted twice.

Table 3: Frequency of the number of admissions per patient

Number of Admissions	Count
1	38983
2	5160
3	1342
4	508
Greater than 4	527

For this task, I consider only the first admission of a patient to determine whether they will be readmitted. For example, if a patient in the dataset has four admissions, the model will only consider the patient's very first admission to the hospital to determine if there will be a second admission. This is because of the fact that few patients have multiple admissions and because considering the first admission of patients alone improves the class imbalance of readmission labels.

2.1.3 Readmission Labels

True or false labels for readmission are generated using data in the `admissions` table [7]. I generate labels for readmission within 30 days, within 90 days, within 365 days, and for overall readmission over any time frame. The readmission label is true if the admission time of a patient's second admission is within 30, 90, or 365 days of the discharge time of their first admission. If a patient has a second admission at all then their overall readmission label is true.

The final datasets contain multiple notes per patient. Each note will have its own set of labels for readmission. If a patient has a true or false value for readmission, each note associated with that patient will have the same labels. Table 4 shows the distribution of readmission labels.

Table 4: Distribution of readmission labels in the general dataset, dataset (1). This contains all notes including discharge summaries for all patients, excluding newborn patients and those who die within a year of their first admission.

Label	Number of Patients	% of Overall Patients
Readmission within 30	1777	5.2%
Readmission within 90	2925	8.5%
Readmission within 365	4570	13.3%
Readmission	7206	20.9%
Overall Patients	34438	100%

The overall number of patients in this dataset is less than the overall number of patients in the MIMIC-III database because a significant portion of patients are filtered out. No newborns are included in the dataset. Furthermore, any patient that dies within a year of their first admission is not included. This is because the readmission value of a patient who dies within a year is arbitrarily false. Moreover, their clinical note data is likely to be significantly different from those of patients who are either healthy or are likely to be readmitted.

2.1.4 Clinical Notes

Null Admission IDs
There are 2,078,705 notes in the `noteevents` table and each note is associated with an admission ID [7]. Of these notes, around 11% – or 231,836 – have a null admission ID. It is therefore difficult to determine exactly which admission such notes originate from. Notes with null admission IDs are referred to as loose notes. To assign an admission ID to a loose note, I look at all the other notes belonging to that patient with non-null admission IDs and assign the ID of the note chronologically closest to the loose note.

Discharge Summary Notes
Each clinical note belongs to one of 15 categories. The distribution of notes in each category is shown in Figure 2. Nursing notes are the most common within

the dataset. Although discharge summaries are a much less frequent category of notes, they are very valuable. Discharge summaries are reports prepared by clinicians at the end of a patient's hospital stay. Studies show that well-documented discharge summaries reduce the likelihood of patient readmission [1]. They also contain valuable summary information that may indicate readmission. As such, they are useful for a model predicting patient readmission. However, including discharge summaries during training changes the clinical relevance of the model. Since discharge summaries are written at the end of a patient's stay, patient care cannot be improved if a model indicates a patient about to leave the hospital might be readmitted. I therefore also consider both tasks of predicting patient readmission with discharge summary notes and without discharge summary notes.

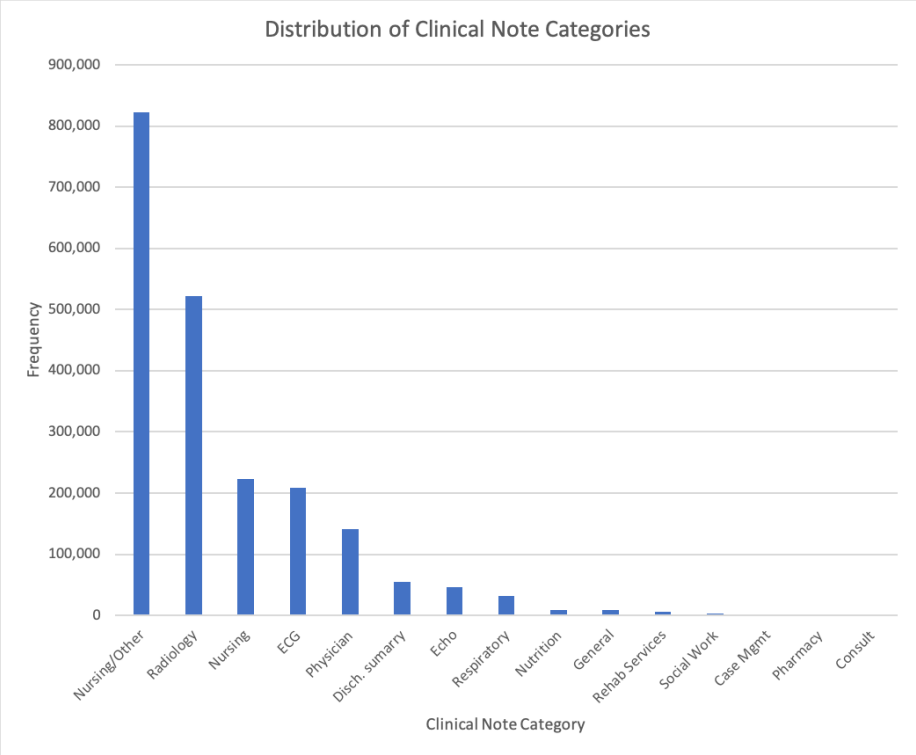


Figure 2: Distribution of clinical note categories in the `noteeveents` table. "Disch. summary" is short for Discharge summary. "Case mgmt" is short for Case management.

The category of the note may also provide some relevant information whilst training the model. All major categories are therefore used alongside the text data when training. Since the categories of "Nutrition," "General," "Rehab Services," "Case management," "Pharmacy," and "Consult" occur so rarely,

they are subsumed under the larger "Nursing/other" category.

2.1.5 Final Datasets

The model is analysed on four final datasets given the above reasoning. They are: (1) all notes including discharge summaries for patients with any condition, (2) all notes excluding discharge summaries for patients with any condition, (3) all notes including discharge summaries for patients with heart failure, (4) all notes excluding discharge summaries for patients with heart failure. Table 4 shows the distribution of labels for dataset (1). The following Tables 5, 6, and 7 show the distribution of labels for datasets (2), (3), and (4) respectively.

Table 5: Distribution of readmission labels in dataset (2). This contains all notes for all patients, excluding discharge summary notes.

Label	Number of Patients	% of Overall Patients
Readmission within 30	1774	5.2%
Readmission within 90	2920	8.5%
Readmission within 365	4562	13.3%
Readmission	7196	20.9%
Overall Patients	34335	100%

Table 6: Distribution of readmission labels in dataset (3). This contains all notes for patients with heart failure, including discharge summary notes.

Label	Number of Patients	% of Overall Patients
Readmission within 30	769	8.5%
Readmission within 90	1292	14.3%
Readmission within 365	2038	22.5%
Readmission	3389	37.4%
Overall Patients	9051	100%

Table 7: Distribution of readmission labels in dataset (4). This contains all notes for patients with heart failure, excluding discharge summary notes.

Label	Number of Patients	% of Overall Patients
Readmission within 30	769	8.5%
Readmission within 90	1292	14.3%
Readmission within 365	2038	22.5%
Readmission	3389	37.4%
Overall Patients	9049	100%

To create the test set, I randomly sample 20% of the patients in the overall dataset. The remaining 80% of patients are split between a training set and a

validation set. This validation set is used to tune the hyper parameters for the model. Once the hyper parameters are selected, the remaining 80% is used to train the model, which is then evaluated on the test set.

As Table 4 shows, there is severe class imbalance in the readmission labels. To counteract this, I down-sample the majority label and up-sample the minority label in the training set. This class imbalance is particularly pronounced for the "readmission within 30 days" and the "readmission within 90 days" labels, and less pronounced for the "readmission within 365 days" and the general "readmission" labels. Therefore, the degree to which the classes are up-sampled and down-sampled differs according to the label.

For the general "readmission" and "readmission within 365 days" labels, the majority class (where readmission is false) is down-sampled until a ratio of 5:1 is reached with respect to the minority class (where readmission is true). Then, the minority class is up-sampled until a ratio of 2:1 is achieved between the majority and minority classes. Similarly, for the "readmission within 30 days" and "readmission within 90 days" labels, the majority case is down-sampled until a ratio of 10:1 is achieved with respect to the minority class. Then, the minority class is up-sampled until a final ratio of 3:1 is achieved between the majority and minority classes. The test set is not up-sampled or down-sampled and therefore retains the same ratio of true to false readmission labels as the original dataset.

2.2 Model

The text in clinical notes is tokenized and any words relating to readmission, death, or survival are removed. This is to prevent the model from learning that a word such as "dead" indicates that a patient will not be readmitted as they will die. Such a prediction would be trivial. Word embeddings are learnt in an unsupervised manner using the gensim library's [14] implementation of the skip-gram word2vec algorithm [12]. The vocabulary consists of the 300,000 most common words throughout all clinical notes where each word must appear at least 7 times across all notes. The learnt word embeddings are then used to train the CNN architecture – WordCNN – described by Zerveas *et al.* [16].

2.2.1 Alternative Models

Bidirectional transformer neural network architectures have demonstrated state-of-the-art performance on natural language processing tasks. These BERT models [3] have been successfully applied to the task of readmission prediction [5]. However, they are best suited for short inputs with similar lengths. Longer sequence inputs result in long runtimes and require the use of costly computational resources. As Zerveas *et al.* discuss, the average length of clinical notes is significantly longer than the sequence length supported by BERT models. The WordCNN model is therefore a promising and less computationally expensive alternative to BERT models.

2.2.2 WordCNN

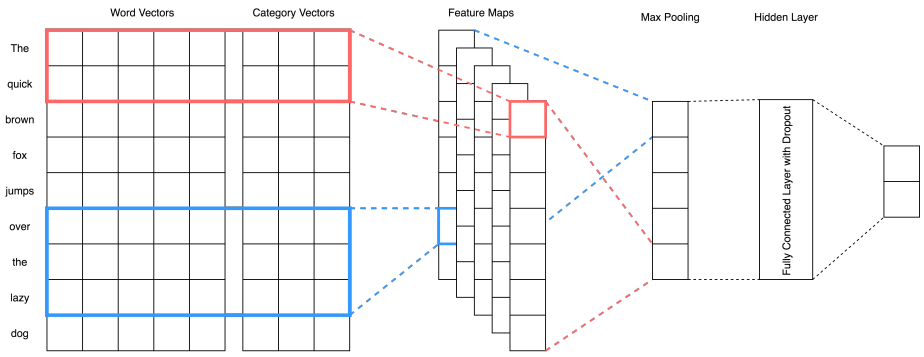


Figure 3: WordCNN architecture. Based on the diagram used by Kim [8].

The WordCNN architecture is based on the CNN architecture described by Kim [8]. Let each word i in an EHR be represented by the word vector $x_i \in R^D$. Let the concatenation, indicated by the operator $\oplus$, of word vectors from i to j be represented by $x_{i:j}$. A patient's EHR containing l words is therefore represented as follows.

$$x_{1:l} = x_1 \oplus x_2 \oplus \ldots \oplus x_l$$

The category of a note is represented by a one-hot vector of length C. Each category vector is appended to its corresponding word vector. Since the category of two notes can differ, the category vector is the only means by which the model can differentiate between two successive notes.

Convolution requires a filter $w \in R^{h \times (C+D)}$ where h is the window size of the filter such that h adjacent words are convolved to create one output. A feature, where b is the bias term, f is the ReLU function, and $\cdot$ indicates element wise multiplication, is calculated as follows.

$$c_i = f(w \cdot x_{i:i+h-1} + b)$$

When applied to all words in the EHR, $x_{1:l}$, this will result in a feature map $c = [c_1, c_2, \ldots, c_{n-h+1}]$. A max-pooling operation is then performed on the feature map to select the maximum feature $\hat{x} = \max c$. This captures the most important information within the feature map.

Note that this process describes the output of some features from a single convolutional filter. The WordCNN model uses many filters which each output one feature. All of the features are then concatenated and passed through a fully connected softmax layer which outputs the probability distribution, $p(y|u)$, over the readmission labels. Here, y is the ground truth readmission label for a patient and u is the vector of features which is input to the fully connected layer. Given m convolution filters, $u = [\hat{c_1}, \hat{c_2}, \ldots, \hat{c_m}]$. In practice, WordCNN uses 450 total filters – 200 filters of window size 3, 150 filters of window size 4, and 100 filters of window size 5.

To avoid over-fitting, a dropout layer is used before the input to the fully connected layer. A fraction of the hidden units in forward propagation are set to zero during this process. In practice, the keep-probability of hidden units is set to 0.9, meaning that around 0.1 of the hidden unit outputs are set to 0.

The model calculates the cross-entropy loss of the predicted label $\hat{y}$ and the ground truth label y given the feature vector u, where y' is the negation of y.

$$l(y, \hat{y}|u) = -(y \log p(y|u)) + (1 - y) \log p(y'|u)$$

Although up-sampling and down-sampling aid with class imbalance, the model also scales the loss of data samples based on whether the class is true or false for readmission. The weight of a data sample i is given by

$$w_i = 1 + \kappa \left(\frac{f_{max}}{f_i} - 1 \right)$$

where f_i is the frequency of the i-th data sample's class, f_{max} is the frequency of the most common class (readmission class of false), and $\kappa \in [0, 1]$ is the class weight factor. The class weight factor is a tuned hyper parameter – 0 means that loss is not re-scaled to compensate for class imbalance whereas 1 means that loss is fully re-scaled. For readmission within 30 days and readmission within 90 days, due to severe class imbalance, I use $\kappa = 1$. For readmission within 365 days and general readmission, I use $\kappa = 0.5$. The total loss for a single training sample is therefore $L = w_i l(y, \hat{y}|u)$. The objective of training is to minimise this loss over all training samples.

3 Results

3.1 Discharge Summary Experiments

Tables 8 and 9 show experiments where WordCNN is trained and evaluated on various target labels. Table 8 shows the results when the model is trained on dataset (1) – patients with any disease where discharge summary notes are included – and Table 9 shows the results on dataset (2) – patients with any disease where discharge summary notes are excluded. The performance of the model evaluated on datasets (1) and (2) is compared in Figure 4.

Table 8: WordCNN model performance on dataset (1) with patients with any disease and including discharge summary notes.

Label	Precision	Recall	AUROC	AUPRC
Readmission within 30	0.920	0.748	0.685	0.108
Readmission within 90	0.863	0.572	0.723	0.198
Readmission within 365	0.829	0.789	0.736	0.290
Readmission overall	0.762	0.716	0.727	0.409

Table 9: WordCNN model performance on dataset (2) with patients with any disease and excluding discharge summary notes.

Label	Precision	Recall	AUROC	AUPRC
Readmission within 30	0.917	0.714	0.646	0.098
Readmission within 90	0.867	0.414	0.714	0.190
Readmission within 365	0.823	0.775	0.718	0.269
Readmission overall	0.760	0.745	0.727	0.416

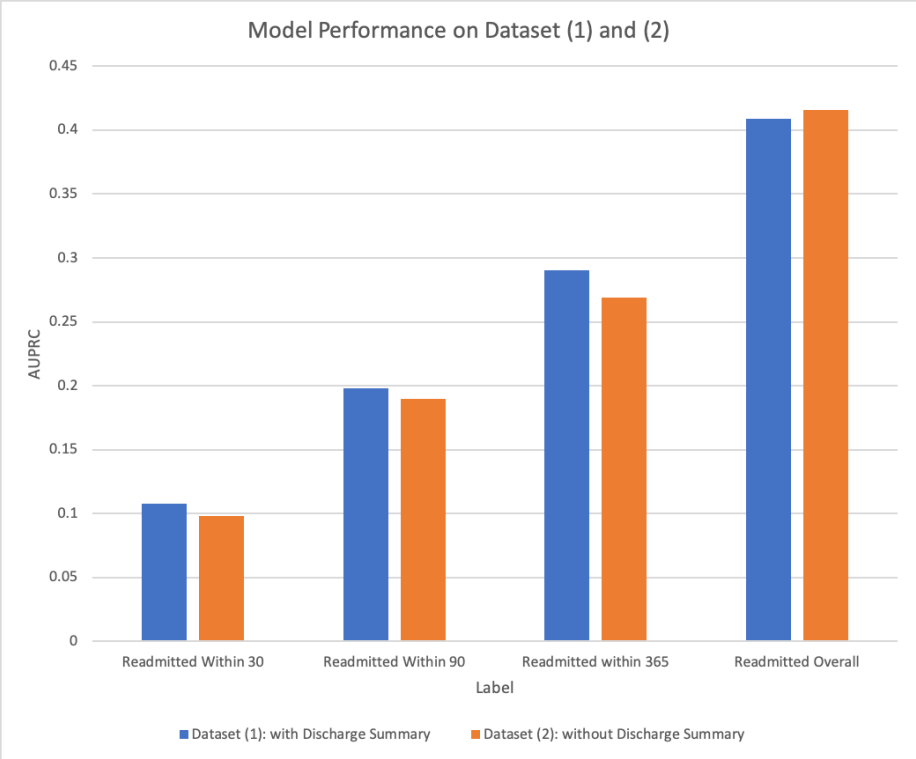


Figure 4: Comparison of model performance when training with dataset (1) – patients with any disease with discharge summary notes – and dataset (2) – patients with any disease without discharge summary notes.

3.2 Model Performance by Parent Diagnosis

Tables 10 and 11 show the model performance when split according to general parent diagnoses groups. These values are generated by analysing the validation data fed into the WordCNN model from dataset (1). The precision and recall are calculated depending on whether the model correctly predicts the label for the validation data. Table 10 shows the results for predicting the overall "readmission" label and Table 11 shows the results for predicting the "readmission within 30 days" label. Both tables include a "Prior Prob." column, indicating the prior probability of readmission. This value is the probability that a patient with a given diagnosis will be readmitted.

For the overall readmission task, the performance of the model is generally correlated with the prior probability of readmission – the higher the prior probability of readmission, the greater the effectiveness of the model in predicting readmission. However, this correlation is weaker in the 30 day readmission task. This possibly indicates that a greater number of positive samples is helpful for training the model and therefore leads to better performance.

Table 10: The precision and recall of the WordCNN model evaluated on validation data for the label of "readmission," split according to the parent diagnosis group of each patient. The "Prior Prob." indicates the prior probability of readmission – the likelihood of readmission for a patient belonging to a particular diagnosis group.

Parent ICD9 Code	Count	Prior Prob.	Precision	Recall
390-459	323	0.079	0.592	0.531
800-999	1534	0.094	0.664	0.499
520-579	947	0.124	0.617	0.643
460-519	864	0.159	0.743	0.689
140-239	705	0.082	0.558	0.606
001-139	672	0.146	0.705	0.686
240-279	267	0.112	0.741	0.717
580-629	206	0.108	0.587	0.657
320-389	176	0.115	0.600	0.609
780-799	135	0.104	0.745	0.612

Table 11: The precision and recall of the WordCNN model evaluated on validation data for the label of "readmission within 30 days," split according to the parent diagnosis group of each patient. The "Prior Prob." indicates the prior probability of readmission – the likelihood of readmission for a patient belonging to a particular diagnosis group.

Parent ICD9 Code	Count	Prior Prob.	Precision	Recall
390-459	3353	0.079	0.189	0.388
800-999	1493	0.094	0.218	0.297
520-579	970	0.124	0.177	0.342
460-519	785	0.159	0.249	0.349
001-139	732	0.146	0.231	0.352
140-239	667	0.082	0.102	0.193
240-279	247	0.112	0.137	0.318
580-629	203	0.108	0.136	0.158
320-389	148	0.116	0.136	0.157
710-739	131	0.062	0.200	0.286

3.3 Heart Failure Experiments

Tables 12 and 13 show the performance of the WordCNN model when evaluated on a limited cohort of patients with heart failure. As outlined in the methodology, some researchers focus on readmission prediction specifically for heart failure patients because the rate of readmission for these patients is higher than for patients with other diseases [10]. This lowers the class imbalance in the dataset, helping the model achieve better performance. Moreover, since heart failure is a very common diagnosis which often requires readmission, the task is clinically relevant.

The model is trained on two heart failure patient datasets, one without discharge summary notes – dataset (3) – and one with discharge summary notes – dataset (4). The comparison of the performance of the model on these two datasets is shown in Figure 5.

Table 12: WordCNN model performance on dataset (3) with patients with heart failure and including discharge summary notes.

Label	Precision	Recall	AUROC	AUPRC
Readmission within 30	0.876	0.697	0.657	0.138
Readmission within 90	0.802	0.646	0.653	0.229
Readmission within 365	0.743	0.754	0.697	0.379
Readmission overall	0.670	0.677	0.705	0.587

Table 13: WordCNN model performance on dataset (4) with patients with heart failure and excluding discharge summary notes.

Label	Precision	Recall	AUROC	AUPRC
Readmission within 30	0.849	0.439	0.527	0.103
Readmission within 90	0.783	0.780	0.649	0.227
Readmission within 365	0.707	0.741	0.663	0.356
Readmission overall	0.668	0.672	0.712	0.598

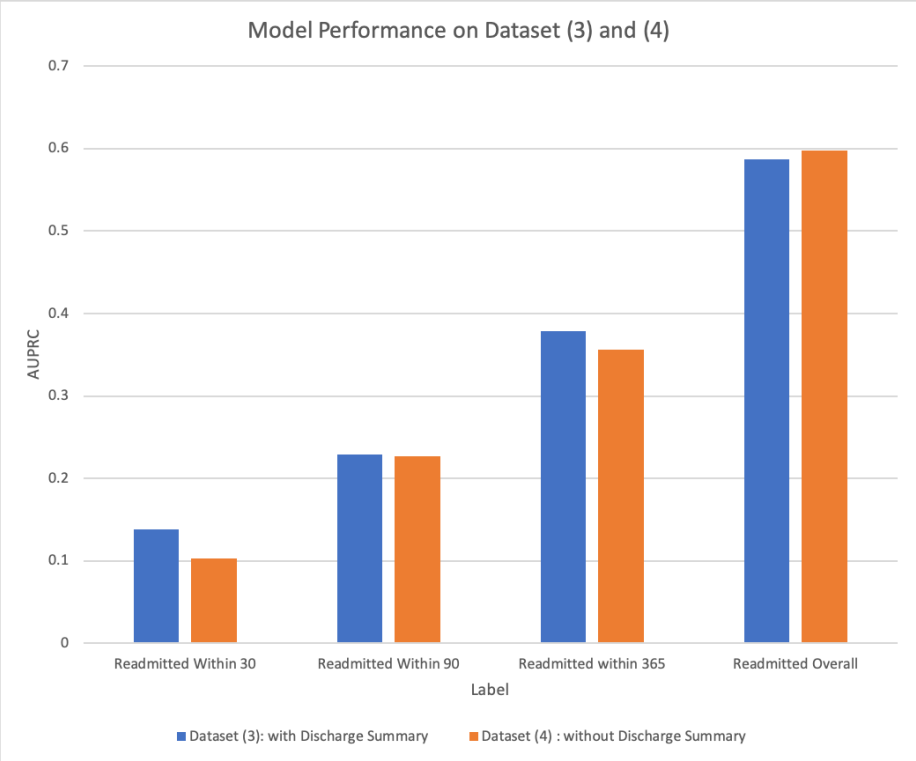


Figure 5: Comparison of model performance when training with dataset (3) – patients with heart failure and including discharge summaries – and dataset (4) – patients with heart failure and excluding discharge summaries.

3.4 Comparison to other models

The WordCNN model is compared to other works on both general readmission prediction and heart failure readmission prediction. The following results are lifted directly from other research papers. Although this provides a point of comparison for the performance of WordCNN, note that the results may not be directly comparable. Johnson *et al.* demonstrate the difficulty of reproducing published results on MIMIC data due to a lack of in-depth reporting on methodology and model design [6]. Although WordCNN is compared to other works with similar methodologies, Johnson *et al.* suggest that an entirely fair comparison is unlikely.

3.4.1 General Patient Comparison

Table 14 shows the comparison between the WordCNN model and the ClinicalBERT model developed by Huang *et al.* which learns deep representations of clinical text [5]. This model achieved state-of-the-art performance in the read-

mission prediction task. ClinicalBERT's superior AUPRC score indicates that it outperforms WordCNN on general readmission prediction. However, WordCNN performs well on the subset of patients with heart failure. This suggests that contextualised word representations, such as ClinicalBERT, are superior in general readmission prediction perhaps because they can discern whether a word indicates readmission in certain contexts only. This may be a less useful feature when a cohort of patients all have similar diseases.

Table 14: Comparison of WordCNN with Huang *et al.* for the 30 day readmission task.

Model	AUROC	AUPRC
Huang *et al.* – ClinicalBERT	0.768	0.747
WordCNN	0.646	0.098

3.4.2 Congestive Heart Failure Comparison

Table 15 and Table 16 show the comparison between results achieved by WordCNN compared to the results achieved by Liu *et al.* [10]. Liu *et al.* use a CNN model as well as another benchmark model, random forests, for both 30 day readmission and general readmission. They include only patients who received care in the ICU and also consider multiple readmissions beyond the first admission.

Table 15: Comparison of WordCNN with Liu *et al.* for the 30 day readmission task.

Model	Precision	Recall	Accuracy
Liu *et al.* – CNN	0.698	0.771	0.719
Liu *et al.* - Random Forest	0.690	0.625	0.672
WordCNN	0.876	0.697	0.697

Table 16: Comparison of WordCNN with Liu *et al.* for the general readmission task.

Model	Precision	Recall	Accuracy
Liu *et al.* – CNN	0.759	0.754	0.757
Liu *et al.* - Random Forest	0.720	0.633	0.694
WordCNN	0.670	0.677	0.677

4 Conclusion

This paper investigated the performance of the WordCNN model proposed by Zerveas *et al.* for the task of readmission prediction. WordCNN leverages unstructured clinical note data to predict readmission over multiple time horizons. The model was evaluated on patients with any diseases as well as a subgroup of patients with heart failure. Moreover, model performance was assessed given datasets both including discharge summary notes and excluding discharge summary notes.

Class Imbalance
The WordCNN model clearly performs better in the global readmission prediction task than the 30 day readmission prediction task. This trend is consistent between all four datasets and when divided by patient diagnoses. The reason behind this trend is the class imbalance in shorter time-frame readmission tasks compared to the longer time-frame or global readmission tasks.

Discharge Summaries
The WordCNN model performs better in the 30 day readmission task using notes including discharge summaries compared to notes not including discharge summaries. This trend is consistent in the 90 day and 365 day readmission tasks, for both patients with any disease and for patients with heart failure. However, discharge summaries appear to make no difference for the general readmission task. In fact, the inclusion of discharge summaries seems to decrease model performance in this case. This may be due to random noise – simply the fact that there are less readmissions in 30 days than overall – or because there is some indication in discharge summaries of whether a patient will be readmitted soon.

These two tasks – training on notes including discharge summaries and excluding discharge summaries – have different clinical relevance. Discharge summaries are only written at the end of stay. Therefore, excluding discharge summaries is necessary if live prediction of readmission whilst a patient is still in hospital is desired. However, post-discharge readmission prediction can still be useful. Post-discharge treatment can vary such that specific attention can be given to patients if the likelihood of readmission after discharge is deemed likely.

Heart Failure Patients
The model achieves far better results on the subset of patients with heart failure than on patients with any disease. Partly, this is likely due to class imbalance. There is less class imbalance in the heart failure patient cohort than in the general patient cohort. However, it is also likely that some underlying similarities between patients with heart failure make the task easier. Again, training a model dependent on disease changes the clinical relevance of the model. Although it can be applied with greater accuracy to patients with heart failure, it does not generalise to apply to many other patients.

5 Further Work

Given the success of the WordCNN model in predicting readmission for heart failure patients, it is yet to be determined if the model performs similarly well for other disease groups. Such research may reveal if there are underlying indicators in notes for heart failure patients which help the readmission prediction task. It is also surprising that the WordCNN model performs significantly worse in the 30 day readmission task for patients with any disease than other similar models. Further research is required to understand the discrepancy between these two sets of results. Furthermore, the current model uses only clinical text data to predict readmission. The inclusion of structured data and unstructured data in the neural network model may improve results.

References

[1] Al-Damluji, M. S., Dzara, K., Hodshon, B., Punnanithinont, N., Krumholz, H. M., Chaudhry, S. I., & Horwitz, L. I. (2015). Association of Discharge Summary Quality With Readmission Risk for Patients Hospitalized With Heart Failure Exacerbation. *Circulation: Cardiovascular Quality and Outcomes, 8(1)*, 109–111. doi: 10.1161/circoutcomes.114.001476.

[2] Boag, W., Doss, D., Naumann, T., & Szolovits, P. (2018). What's in a Note? Unpacking Predictive Value in Clinical Note Representations. *AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science, 2017*, 26–34.

[3] Devlin, J., Chang, M-W., Lee, K., Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:1810.04805*.

[4] Futoma, J., Morris, J., & Lucas, J. (2015). A comparison of models for predicting early hospital readmissions. *Journal of Biomedical Informatics*, 56, 229–238. doi: 10.1016/j.jbi.2015.05.016.

[5] Huang, K., Altosaar, J., & Ranganath, R. (2019). ClinicalBERT: Modeling Clinical Notes and Predicting Hospital Readmission. *arXiv preprint arXiv:1904.05342*.

[6] Johnson, A.E.W., Pollard, T.J. & Mark, R.G. (2017). Reproducibility in critical care: a mortality prediction case study. *Proceedings of the 2nd Machine Learning for Healthcare Conference, in PMLR* 68:361-376.

[7] Johnson, A.E.W., Pollard, T.J., Shen, L., Lehman, L., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L.A., & Mark, R.G. (2016). MIMIC-III. *Scientific Data* . DOI: 10.1038/sdata.2016.35.

[8] Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. *arXiv preprint arXiv:1408.5882*.

[9] Kommers, A.-M. (2019, June 18). Average hospital readmission costs for 18 diagnoses. Retrieved from https://www.beckershospitalreview.com/rankings-and-ratings/average-hospital-readmission-costs-for-18-diagnoses.html.

[10] Liu, X., Chen, Y., Bae, J., Li, H., Johnston, J., & Sanger, T. (2019). Predicting Heart Failure Readmission from Clinical Notes Using Deep Learning. *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. doi: 10.1109/bibm47256.2019.8983095.

[11] McIlvennan, C. K., Eapen, Z. J., & Allen, L. A. (2015). Hospital readmissions reduction program. *Circulation, 131(20)*, 1796–1803. doi: 10.1161/CIRCULATIONAHA.114.010270.

[12] Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J., (2013). Distributed Representations of Words and Phrases and their Compositionality. *arXiv preprint arXiv:1310.4546*.

[13] Rajkomar, A., Oren, E., Chen, K., Dai, A. M., Hajaj, N., Hardt, M., Liu, P. J., Liu, X., Marcus, J., Sun, M., Sundberg, P., Yee, H., Zhang, K., Zhang, Y., Flores, G., Duggan, G. E., Irvine, J., Le, Q., Litsch, K., Mossin, A., ... Dean, J. (2018). Scalable and accurate deep learning with electronic health records. NPJ digital medicine, 1, 18. https://doi.org/10.1038/s41746-018-0029-1.

[14] Řehůřek, R., Sojka, P. (2010). Software Framework for Topic Modelling with Large Corpora. In *Proceedings of LREC 2010 workshop New Challenges for NLP Frameworks*. 46-50. ISBN 2-9517408-6-7.

[15] van Walraven, C., Bennett, C., Jennings, A., Austin, P. C., & Forster, A. J. (2011). Proportion of hospital readmissions deemed avoidable: a systematic review. *Canadian Medical Association Journal*, 183(7). doi: 10.1503/cmaj.101860.

[16] Zerveas, G., Schmidt, F., Eickhoff, C. (2018). Neural NLP methods for online clinical decision support. Manuscript submitted for publication.

[17] Zhang, Y., Jin, R. & Zhou, Z. Understanding bag-of-words model: a statistical framework. *International Journal of Machine Learning & Cybernetics*. 1, 43–52 (2010). https://doi.org/10.1007/s13042-010-0001-0.