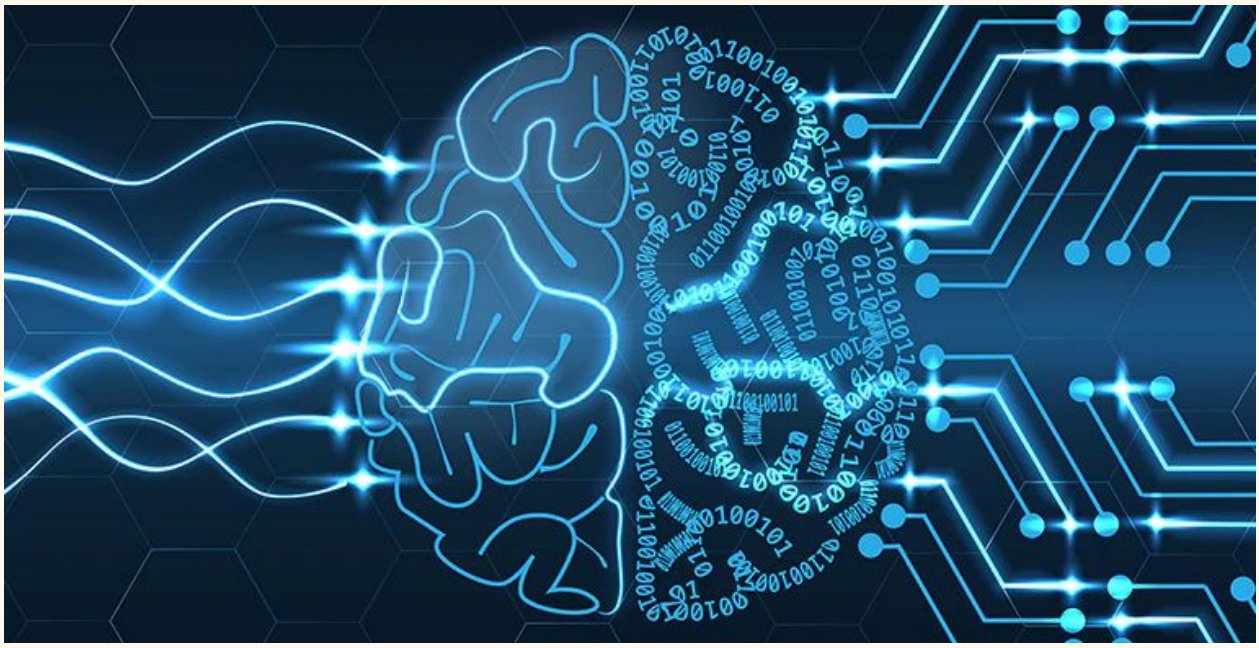
CIINICAL
BIG DATA RESEARCH

Lelei Zhang, Muyao Chen, Shiying Tu, Bin Tong, Zhixuan Chi



INTRODUCTION

Since a person born from a hospital, medical professionals start to record every detail in clinical notes. Those notes represent a vast wealth of knowledge and insight that can be utilized by data scientists to improve patient care and hospital workflow. BC spends approximately 1 billion dollars each year to look for a way to sort out large amounts of clinical notes and charts[1]. With the huge amount of data inspiring research and development, Big data solutions to optimize clinical care and improve patient outcomes across the care continuum.

Motivation and Background

Big Data should not only be limited to bank and tech companies, instead it should be applied to the field that can bring social benefits to the public. Today, the majority of people do not suffer from a shortage of water and food, but medical resources. Our team wondered if the data science skills could help on several clinical tasks to help doctors make a smart decision. They

are “Medical Entity Recognition”, ”Mortality Rate Prediction”, “Length of Stay Prediction”, and “Readmission Prediction”. The good news is that we have a lot of different data sources from MIMIC III to analyze, and the bad news is that by far there are no end-to-end solutions in clinical usage since it requires a very higher accuracy and the model must be strongly explainable. In this project, we will mainly focus on exploration on data, transplantation of existing models, and solutions to unknown challenges.

Problem Statement

Medical Entity Recognition

Our first question is to extract medical entities such as chemical and disease names from unstructured clinical notes. Medical Name Entity Recognition(NER) has become crucial in the current NLP field since extracted entities can help doctors treat patients and minimize medical errors. Patients can recognize the types of medical entities without any knowledge background.

Despite its importance, the task is challenging to work on. Firstly, the real datasets are not possible to share with students without legal agreement given the sensitivity and privacy in healthcare records. By following the suggestions, we use the MIMIC-III critical care database. However, the database lacks information about medical tags and thus supervised learning becomes an impossible option for this task. Secondly, the topic focuses on the medical field, general NLP models may suffer from uncovering high-quality relationships between medical concepts as judged by humans. A model specific to the clinical field needs to be implemented.

Mortality Rate Prediction

In our project, we have considered diverse application scenarios. On the one hand, we can handle newly generated clinical notes using model builds on unstructured data, on the other hand, we can take advantage of existing complicated medical databases using models based on the structured data.

The mortality rate prediction is a valuable medical indicator that can assist doctors to make an appropriate treatment plan. For some diseases such as COVID-19, the patient’s condition can suddenly be changed to a fatal situation with no early-day omen. A Machine Learning model, which takes various medical details into account, can help doctors to detect in advance.

Length of Stay Prediction

We also covered the clinical problems of forecasting length of stay (LOS). Each person's LOS is defined based on the number of days between their admission and discharge from the hospital.

Due to the expensive cost of hospital stays, which adds up to 400 billion dollars annually in the United States. Accurately predicting LOS can reduce the risk of infection and side effects of medication, improve the quality of treatment and increase hospital profits.

Medical issues are sensitive, rigorous calibration of prediction and fair evaluation of predictive accuracy is important for our model optimization.

30-Day Hospital Readmission Prediction

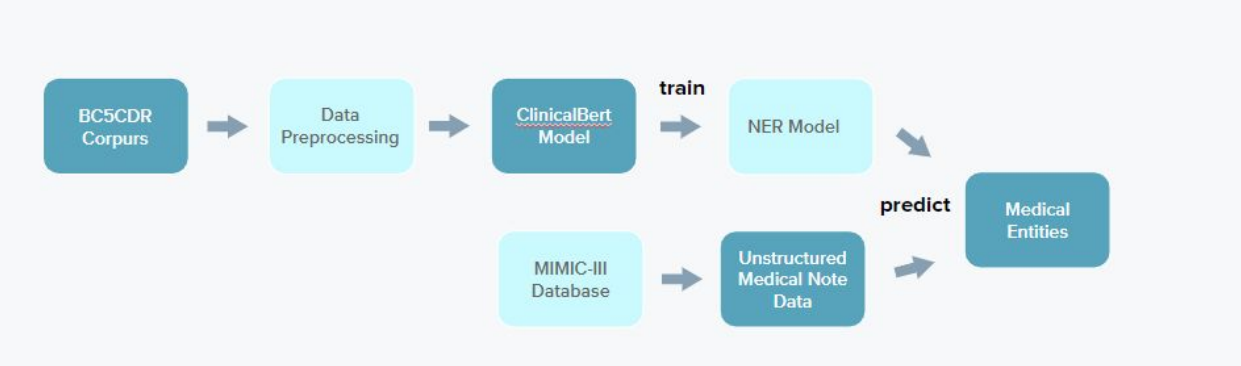
In the final part, we focus on the task of building a model for hospital readmission prediction using clinical notes. Readmission nowadays is a trusted measure of effective and responsive care. Knowing the patients' readmission status in the next 30-day can help hospitals to adjust various plans for better service.

Preprocessing the data to get a 30-day readmission label is a complicated, yet crucial step. For example, in MIMIC-III, 5,854 admissions are in-hospital deaths. Since deaths do not imply readmission, those admissions should be removed. The same happens to newborns. Secondly, each patient may have multiple admissions and notes. Training on all of them for patients is a lot of work. In this project, we find an effective way to predict readmission using only early notes for patients. Thirdly, the MIMIC-III dataset has an imbalanced readmission rate. Readmitted in 30-day vs not Readmitted is roughly 3:1. The evaluation metric, accuracy, is meaningless in this case and thus other metrics need to be found to evaluate the model.

Data Science Pipeline, Methodologies and Evaluation

Each question has a different workflow. We will describe the data science pipeline by questions.

Medical Entity Recognition



1. Data Collection

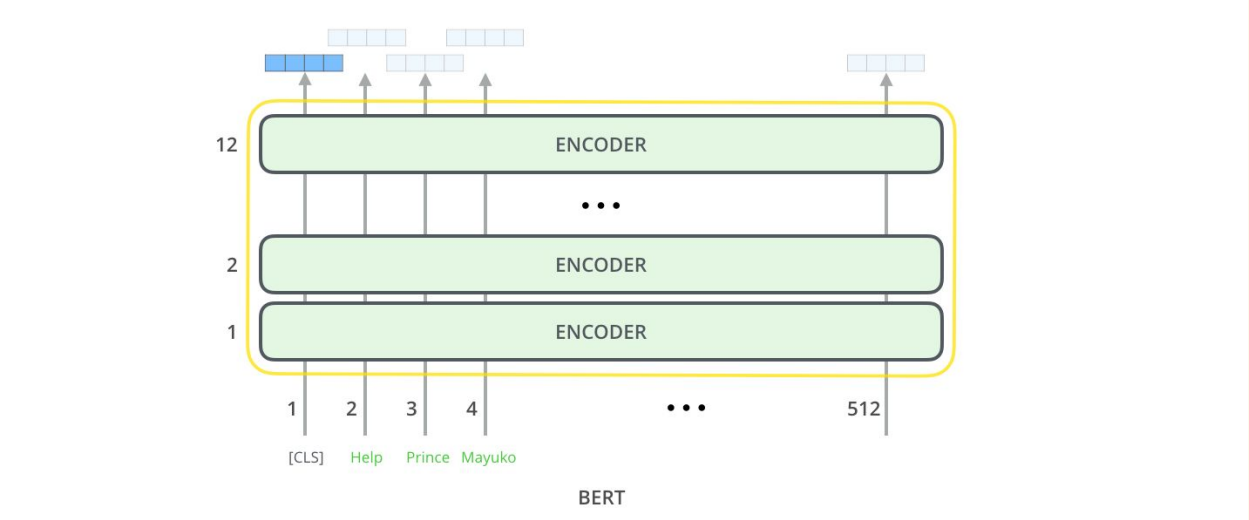
Throughout the whole project, we use the MIMIC-III critical care database, which includes 60,000 intensive care unit admissions, demographics, vital signs, laboratory tests, medications, and more. However, it lacks chemical and disease tags information for clinical notes. Thus we turn to BioCreative V CDR corpus, consisting of 1500 PubMed articles with 4409 annotated chemicals, 5818 diseases, and 3116 chemical-disease interactions.

2. Data cleaning

MIMIC-III is a relational database consisting of 26 tables. In the NER task, we mainly focus on extracting clinical notes, which are stored in the 'TEXT' column of the NOTEEVENTS table. BC5CDR corpus data is stored in tab-separated tsv form. We switch it to space separated since it is required for running the ClinicalBert NER script. Moreover, we filter out some tags for training data since we are performing a singular tag NER. Our final step is to split the data into training, validation and testing set, preparing for the modeling part.

3. Data Integration and Modeling

We use BCD5CDR for the whole training process, and thus there is not much data integration to be done. After careful research and consideration, we believe that ClinicalBert, clinical deep bidirectional transformers, is the best model for medical entities recognition. It uncovers high-quality relationships between medical concepts as judged by humans.



Source: http://jalammar.github.io/illustrated-bert/

ClinicalBert[2] is trained by applying the BERT (Bidirectional Encoder Representations from Transformers) model on clinical notes. The BERT models' architecture is a multi-layer bidirectional Transformer encoder, which has large feedforward-networks with 768 hidden units, and 12 attention heads. As indicated by the above graph, a CLS token is added to each input sequence to the Bert model. The final hidden state corresponding to this token is used as the aggregate sequence representation for classification tasks.

While pre-training the BERT model, two unsupervised tasks: Masked LM and Next Sentence Prediction are performed. Standard conditional language models can only be trained left-to-right or right-to-left to achieve the bidirectional purpose. This training method is relatively shallow since each word is not exposed to the whole context, and thus could trivially predict the target word in a multi-layered context. In order to train a deep bidirectional representation, Bert model simply masks some percentage of the input tokens at random and then predicts those masked tokens. The Bert model also pre-trains for a binarized next sentence prediction task to learn embeddings that better understand sentence relationships.

Turning to our medical entity name recognition application, we perform supervised learning using the pre-trained ClinicalBERT model. The inputs are sentences in BC5CDR with each word tagged as one of B-Disease, I-Disease, B-Chemical, and I-Chemical and the outputs are predicted tags corresponding to each word in the sentences. We use the pre-trained ClinicalBert model plus a softmax classifier as our model. During the

training process, the model learns the text embedding by continuously improving the accuracy of predicted tags.

4. Methodologies

The methodology we used for these two tasks is transfer learning. Humans don't learn everything from the ground up and leverage and transfer their knowledge from previously learnt domains to newer domains and tasks. Transfer learning adopts the same idea that learning a new task relies on previously learned tasks.

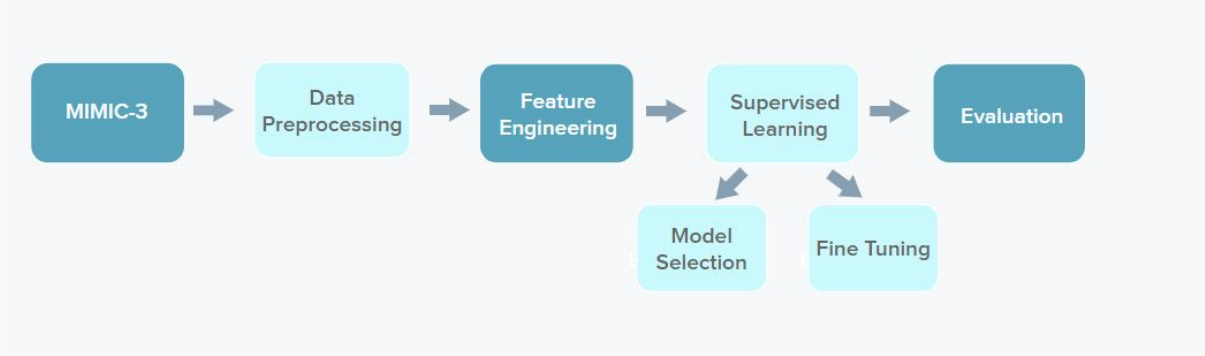
In this project, we adopt the idea of transfer learning and train models very efficiently. The downloaded ClinicalBert model is pre-trained using MIMIC-III dataset. We use additional information such as medical and disease tags and readmission labels to fine-tune the pre-trained model so that it can new embeddings based on learned knowledge for supervised predictions.

5. Evaluation and Visualization

Chemical-f1 = 0.9915183148476093
Chemical-precision = 0.9882943143812709
Chemical-recall = 0.99476341873948
Disease-f1 = 0.9791666666666666
Disease-precision = 0.9730169806931844
Disease-recall = 0.9853945818610129
f1 = 0.9860426503398536
loss = 0.008468479252756914
precision = 0.9815101745687429
recall = 0.9906171809841534

As can be seen from the above statistics, the model reached quite high precision in both chemical and disease predictions. For visualizing the predicted results, we designed a web application, where words are tagged as disease highlighted in yellow and chemicals highlighted in blue.

Mortality Rate Prediction



1. Data Preprocessing.

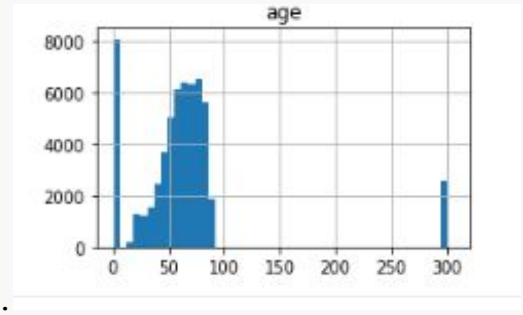
1.1. Data Integration.

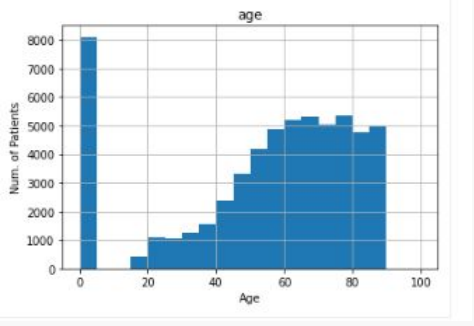
The MIMIC-3 is a huge medical database that includes 26 tables, more than 1 million rows of contents and hundreds of features. The total decompressed size is more than 40 Gigabytes. There exist some large tables which have a size of 10 Gigabytes. In this case, using Pandas and Featuretools to process the data and select the features will be highly time-consuming and space-consuming. Therefore, data integration has been done to summarize the MIMIC-3 database into one relatively small table using a medical book *"Medical Information Extraction & Analysis: From Zero to Hero with a Bit of SQL and a Real-life Database*[5]" as a reference in Spark. The text features such as Callouts, Procedures, Transfers, Micro Labs have been summarized into numerical values on a daily basis. The important categorical and numerical features have been extracted and calculated(decoded). Some of the information in the database has been encoded. For example, the age of each patient has to be decoded by subtracting DOB(Date of birth) from the first INTIME(transfer-in time). We use HADM_ID(admission ID) and SUBJECT_ID(patient ID) join keys to summarize the database into one table with 58976 admission records and 26 features. The summarized table is saved as a .csv file(12MB) that can be processed following with pandas.

```
Data columns (total 26 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   admit_type        58976 non-null  object
 1   admit_location    58976 non-null  object
 2   insurance         58976 non-null  object
 3   religion          58518 non-null  object
 4   marital_status    48848 non-null  object
 5   ethnicity         58976 non-null  object
 6   AdmitDiagnosis    58951 non-null  object
 7   ExpiredHospital   58976 non-null  int64
 8   LOSdays           58976 non-null  float64
 9   age               58976 non-null  float64
 10  gender            58976 non-null  object
 11  NumCallouts       58976 non-null  float64
 12  NumDiagnosis      58976 non-null  float64
 13  NumProcs          58976 non-null  float64
 14  NumCPTevents      58976 non-null  float64
 15  NumInput          58976 non-null  float64
 16  NumOutput         58976 non-null  float64
 17  NumLabs           58976 non-null  float64
 18  NumMicroLabs      58976 non-null  float64
 19  NumNotes          58976 non-null  float64
 20  NumProcEvents     58976 non-null  float64
 21  NumTransfers      58976 non-null  float64
 22  NumChartEvents    58976 non-null  float64
 23  NumRx             58976 non-null  float64
 24  AdmitProcedure    50944 non-null  object
 25  TotalNumInteract  58976 non-null  float64
```

1.2. Restore the blurred information.

During the EDA, we observed a group of patients with age over 300 years old. By doing research, we found this is caused by “DOB has been shifted 300 years before their first admission for patients over 89 years old”. We restored those ages back to 89.





.

1.3. Outlier Detection

The descriptive statistics showed the existence of negative numerical values in the summarized table. This is caused by some negative LOSdays(Length of Stays). As there are only 96 records containing negative LOSdays we just deleted them as outliers.

1.4. Fill the Missing Values

During the summarization, the missing numerical values have been filled with 0 and the missing text has been filled with NA.

2. Feature Engineering

2.1 Feature Selection

The main problem we dealt with in model selection is “Leaking Data From the Future”. When asked to predict mortality, the LOSDays is not yet known, so it should not be given to the model during training. However, the numerical features we calculated using LOSDays are not leaking data since the daily basis values are observations that can be updated by doctors in real life. So we drop the LOSDays from the data and some other irrelevant features according to the research we did.

2.2 Feature Scaling

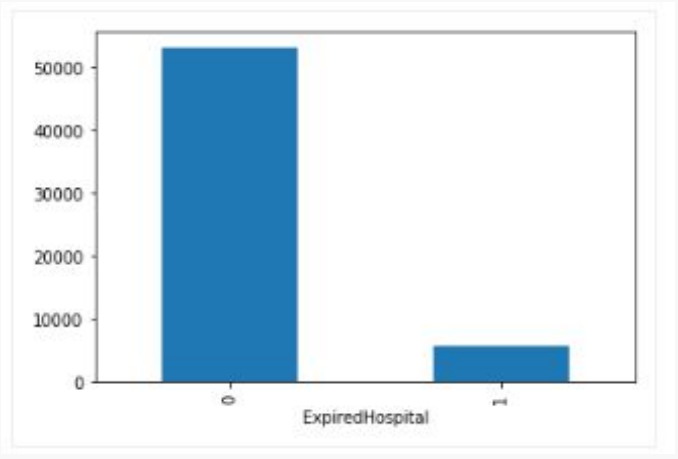
The EDA showed the data skewness, so we normalize the whole dataset.

2.3 Feature Transformation

The categorical data “gender”, “admit_type”, “admit_location” have been transferred to numerical data using one-hot encoding. After the feature engineering, we have 30 columns in total.

3. Supervised Learning

The Visualization of the label(ExpiredHospital) showed the imbalance between label 1 and label0 with ratio 9:1. In this case, the usual evaluation metric accuracy is trivial since the TN observations are high. By doing research and combining the knowledge we learned in class, we decided to use precision, recall, F1 score, ROC-AUC curve, and precision-recall curve as our evaluation metrics. The critical point of those metrics is that they only consider TP.



As we know there is a trade-off between recall and precision. The recall represents the model sensitivity in other words "What proportion of actual positives was identified correctly". In the medical field, we want our model to have higher sensitivity since we are dealing with lives. However, the lower precision rate caused by trade-off could generate a mass of FP(false positive) which could waste the precious medical resources on testing and treatment. Therefore, we tried to keep our recall and precision with a reasonable threshold during our training.

The modeling part has used *Alexander Scarlat, MD 's Machine Learning Primer for Clinicians–Part 8* [4] as reference.

3.1 Model Selection for classification models.

We have selected models from RF, SGD, LogReg, K-nearest, SVM, GaussianNB, Decision tree using cross-validation with evaluations mentioned above. The Random Forest model has a better performance than others with all evaluation metrics.

3.2 Fine Tuning

We tuned the hyperparameters using Grid search.

3.3 Neural Network Model Structure.

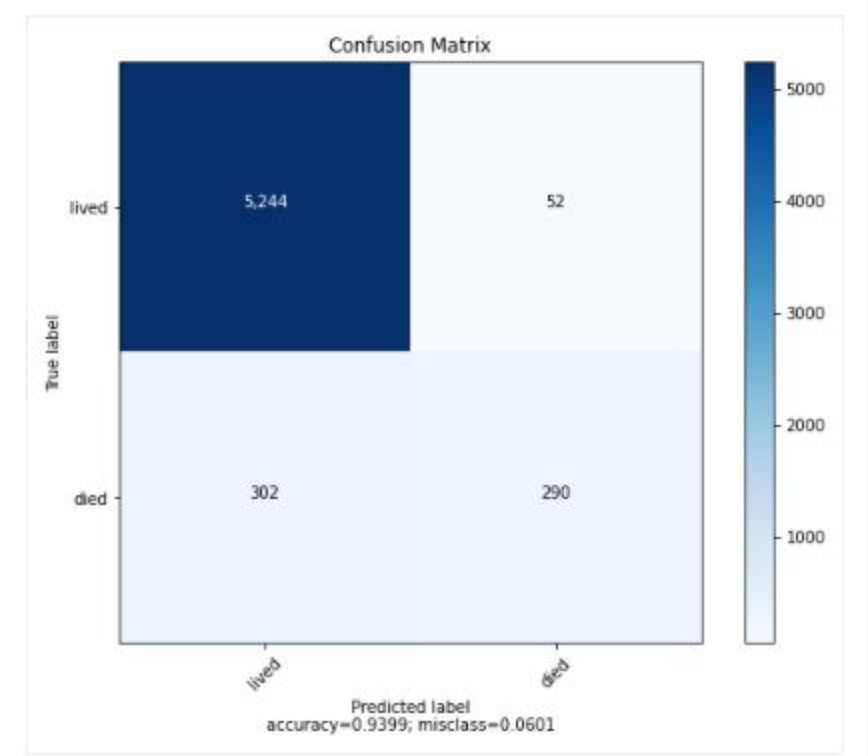
A neural network model is also built to compare with the random forest

model. The NN model includes three hidden layers, each contains 2048 hidden units. The activation function we chose is ReLU. We also added two dropout layers to reduce overfitting.

4. Model Evaluation

4.1 Confusion Matrix

The confusion matrix described the performance of the classifier on the test set.



4.2 Precision, Accuracy, Recall, F1 Score

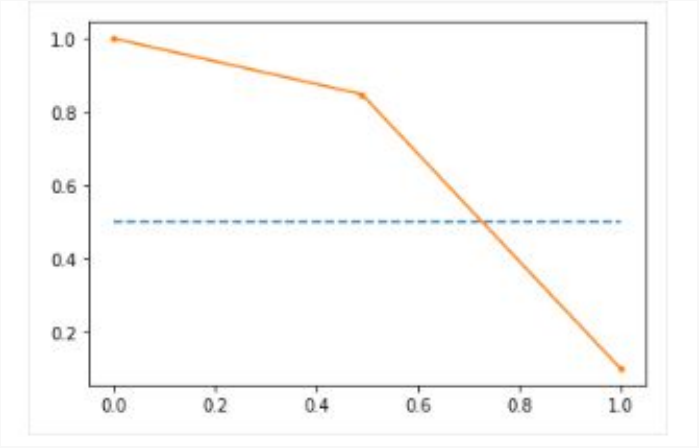
```
precision  0.848
recall  0.4899
accuracy  0.9399
F1 score  0.621
```

4.3 ROC-AUC curve

The closer the curve to the left-up corner, the better performance the model has. In other words, we want to maximize the area under the curve.

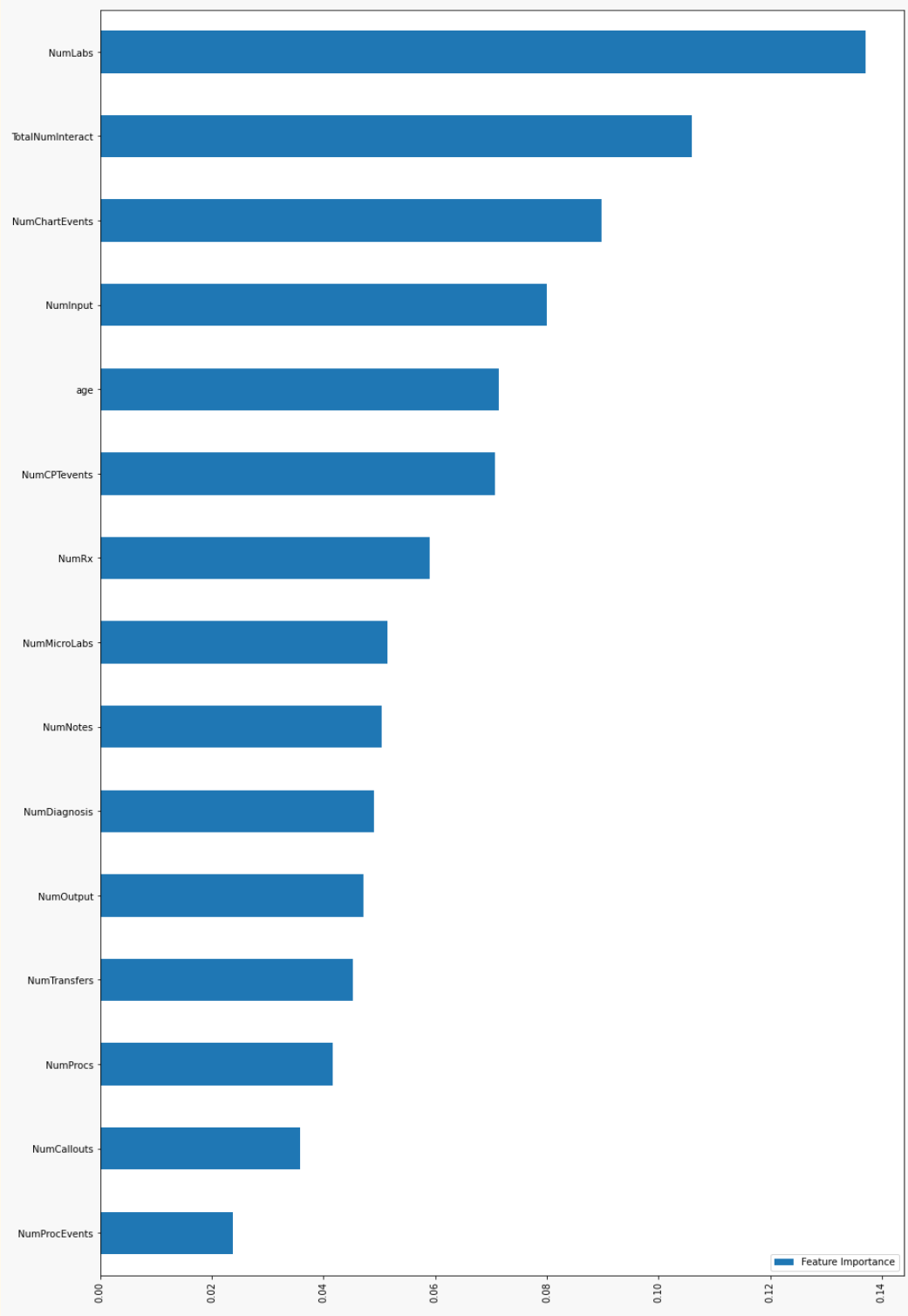
4.4 Precision-Recall curve

Similar to the ROC-AUC curve but the right-up corner.



4.5 Feature Importance

We have visualized the feature importances. The Daily number of labs is the most important feature in our model. The advantage of doing feature selection and data preprocessing with domain knowledge has reflected. We can easily explain the features' meaning here.



Similar evaluations have been done to the NN model.

Length of Stay Prediction

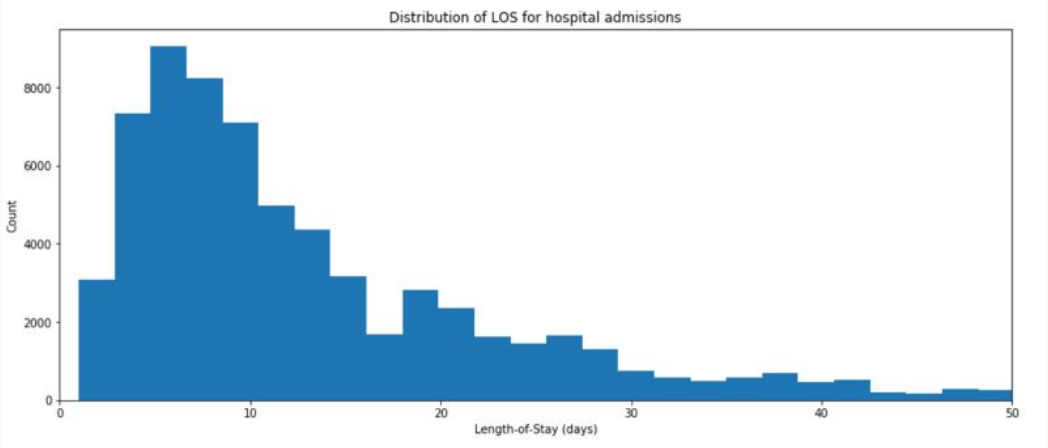
1. Data Preprocessing

We derived data from publicly available intensive care medical information market (MIMIC-III) databases to explore supervised statistical learning models to predict length of stay (LOS). MIMIC-III is a relational database consisting of 26 tables.

Five tables are used to define and track patient stays: ADMISSIONS, PATIENTS, ICUSTAYS, SERVICES and TRANSFERS and seven tables contain data associated with patient care, such as medication prescription, physiological measurements and caregiver observations. After reviewing the contents of each table in the database, we selected 12 tables and performed interpretations based on these tables.

We followed the end-to-end data science pipeline. Conducted initial data cleaning and exploratory data analysis. Imputed missing values, restored blurred information, detected outliers and converted the categorical variables into numeric columns that accepted ML models. We removed patients who resulted in death as it's not a typical impact of length of stay.

The Length of stay is naturally a skewed distribution in most cohorts of patients, as shown below.

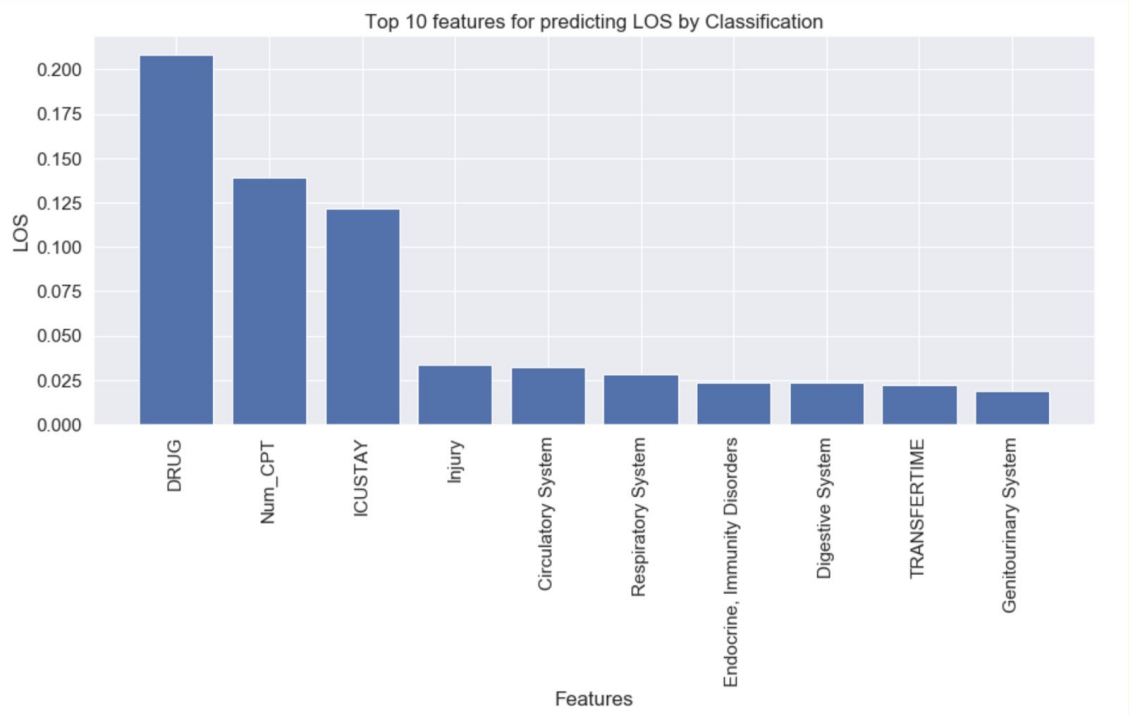


2. Feature Engineering

By incorporating multiple types of features, including age, diagnosis, drug, CPT test, and ICD9 embedding, we found ICD9 (health care classification system) is used in diagnosis for classifying diseases. Approximately 7000 unique codes were used in the dataset and 0.6 million ICD9 diagnoses were given to patients, as each patient may have more than one disease. We reduced the ICD-9 codes from 7000 to 16, sorted all the unique codes per admission into different categories and transformed into the ML models. This ICD9 part has used *Daniel Cummings' Predicting hospital length-of-stay at time of admission*[3] as reference. The subsequent feature importance shows that diagnosis categories are the important features in predicting LOS.

For the remaining features, one-hot encoding was used to convert categorical variables to numerical columns. We also investigated the clinical impact of patients by counting the current procedural terminology (CPT) and number of days each patient stays in ICU.

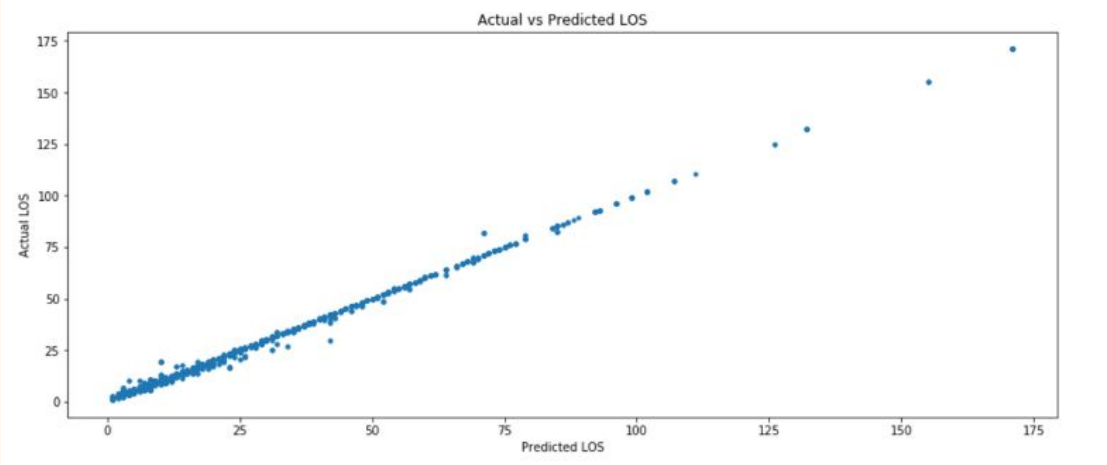
In addition to a number of drugs, CPT test, ICU stays, and disease diagnosis that have large impact on the LOS. The majority of the features seem to have less importance to our model. Specifically, the number of drugs taken by each patient contributed the most to our model, showing a strong positive correlation with the LOS, followed by the number of CPT tests. This means medication prescription is a dominant aspect in health care, the potentially inappropriate medications are essential especially with elderly patients. An issue of inappropriate medications during hospitalization of acutely patients may increase their total length of stays.



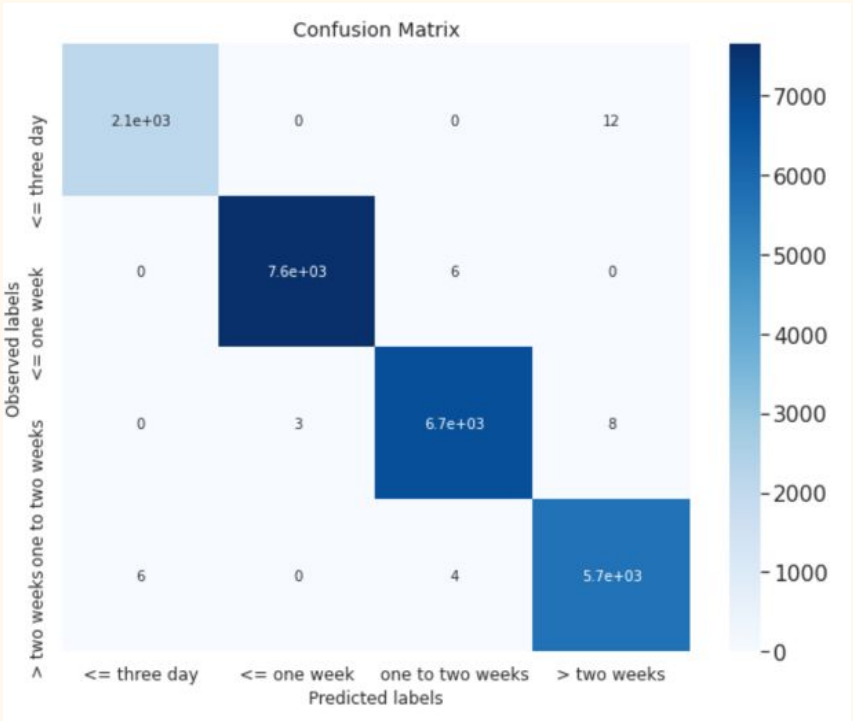
3. Prediction, Methodologies and Evaluation

For prediction, while the mean and median LOS is useful from a certain perspective, it is a poor statistic in terms of representing a typical LOS. Therefore, our models are successfully created to predict the length of stay. Three modelling techniques were used: Regression, Classification and Recurrent Neural Network. The data was randomly split into 80% admissions to derive the model and the remaining 20% admissions to test the model's application.

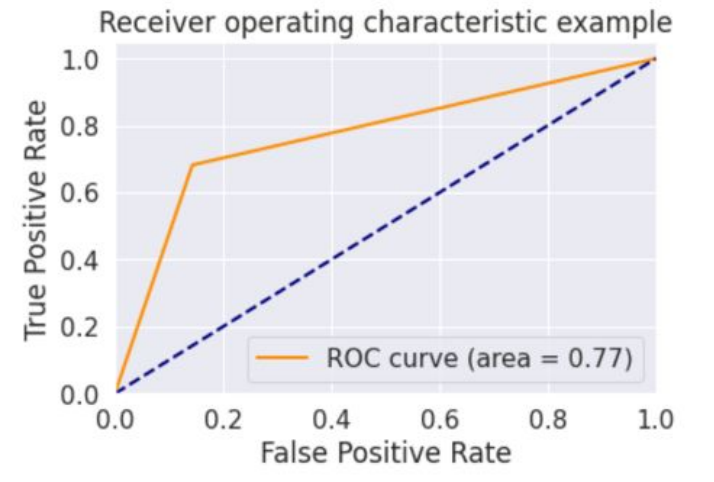
R-Square, Root Mean Square Error (RMSE) and the residual plot were used to check the model fit after implementing five different regression models. In addition, we tuned the parameter on the selected model to improve the performance. From the results, The Random Forest Regression model is prominent. We have reached 99% R-Square and only 0.27% Root Mean Square Error.



Considering the sensitivity in the medical field and avoiding giving the exact number of LOS, we framed the LOS as a classification problem with 4 classes. One less than three days, one from three days to one week, one from one week to two weeks and one more than two weeks. We fitted the training data to multiple classification models and used a confusion matrix to determine the performance of each model. We find out that Random Forest Classification is appropriate for skewed data. The figure below shows the summary of correctly predicted numbers broken down by each class. The classification accuracy is nearly 100 percent.

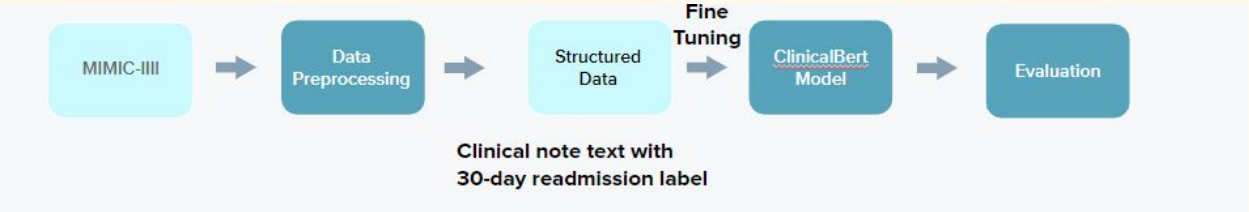


We also utilized a Recurrent Neural Network based on Long Short-Term Memory (LSTM) to construct a model for the prediction of length of stay. We performed the analysis of feature contributions to the predictive model and our model improved the area under the ROC curve to 0.77. However, the recurrent neural network does not perform well compared with the traditional classifier. This is due to the limitation of the sensitive training set and data imbalance, even the integration of ML classifiers cannot overcome this problem.



In conclusion, based on the accuracy of the algorithm using the train test set, Random Forest Regression and Classification provide us with the best result for predicting LOS. Since length of stay is an important indicator of hospital management efficiency. The machine learning models we built can accurately predict the length of stay and be able to help improve the accuracy of hospital decision making, allocation of healthcare resources, patient counseling and hospital administration.

30-day Readmission Prediction



1. Data Collection and Data Cleaning

This part will be supplementary research on pure text processing just in case if in the circumstance the hospital only has the clinical notes available.

In our final task, we prepare the data for training the model by calculating the binary readmission label in 30 days from the MIMIC-III Admission table. Patient admissions for which the patient is readmitted within 30 days receive a label of ***readmit*** = 1. All other patient admissions receive a label of zero. Firstly, we remove the admissions with type 'NewBorn' and 'Death' since these two types are not in our project interests. Secondly, days until the patient's next admission is calculated and we name the field as ***'DAYS_NEXT_ADMIT'***. Thirdly, we mark the admissions with ***'DAYS_NEXT_ADMIT'*** longer than 30 as ***readmit = 0*** and others as ***readmit = 1.***

Each admission is associated with a note stored in the NOTEEVENTS table. We merge these two tables by ***'PATIENT_ID'*** and ***'HADM_ID'***. As a result, each note has its corresponding readmission label. We noticed a patient can have multiple notes. In this task, we only take the first 72-hour notes for each patient. The processed input data to the model is now clinical notes, each with a 30-day readmission label.

2. Data Integration and Modeling

Data integration is not needed for this task since we use MIMIC-III for the whole process. Similar to the medical entity recognition task, we perform supervised learning using the pre-trained ClinicalBert model plus a linear classifier layer. The model learns the embeddings by continuously improving the accuracy of predicting the readmission label for each patient admission.

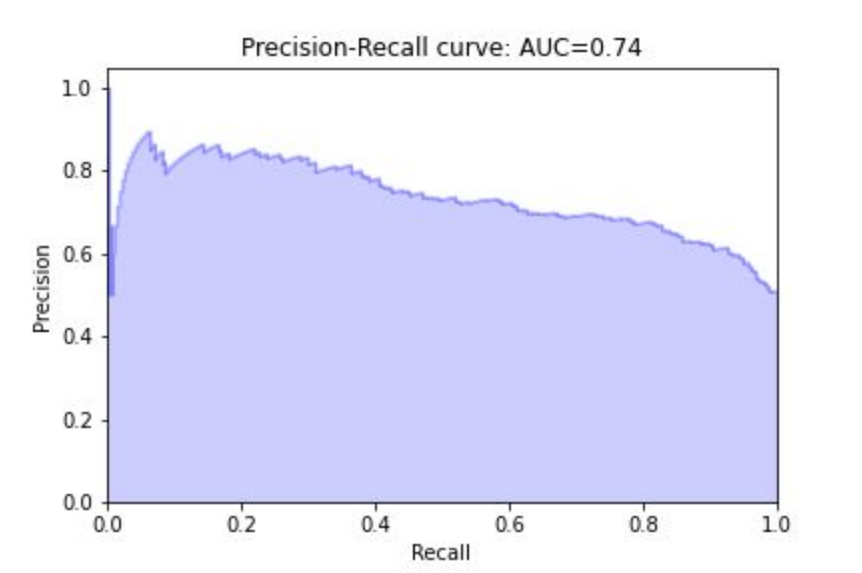
In addition, we implemented our own word vector + LSTM model. We loaded pre-trained word vector models and converted words in each sentence to numerical vectors. After that, we created our own LSTM model to perform a binary classification since the LSTM model is a good solution for sequential inputs. We picked the threshold that maximized the AUC score on the validation dataset. The result showed we reached out a 0.56 accuracy on the test dataset after 6 iterations with the simplest Glove embedding.

This result was at least better than a random classifier. Since we only trained 6 iterations with the easiest embedding (a lot of zero embeddings) and LSTM models, and only 8% dataset, this result was fair to be acceptable since NLP tasks require huge memory space. If the computational resources are more available, we will train a huge word embedding model and applied more complex network structures. In addition, the

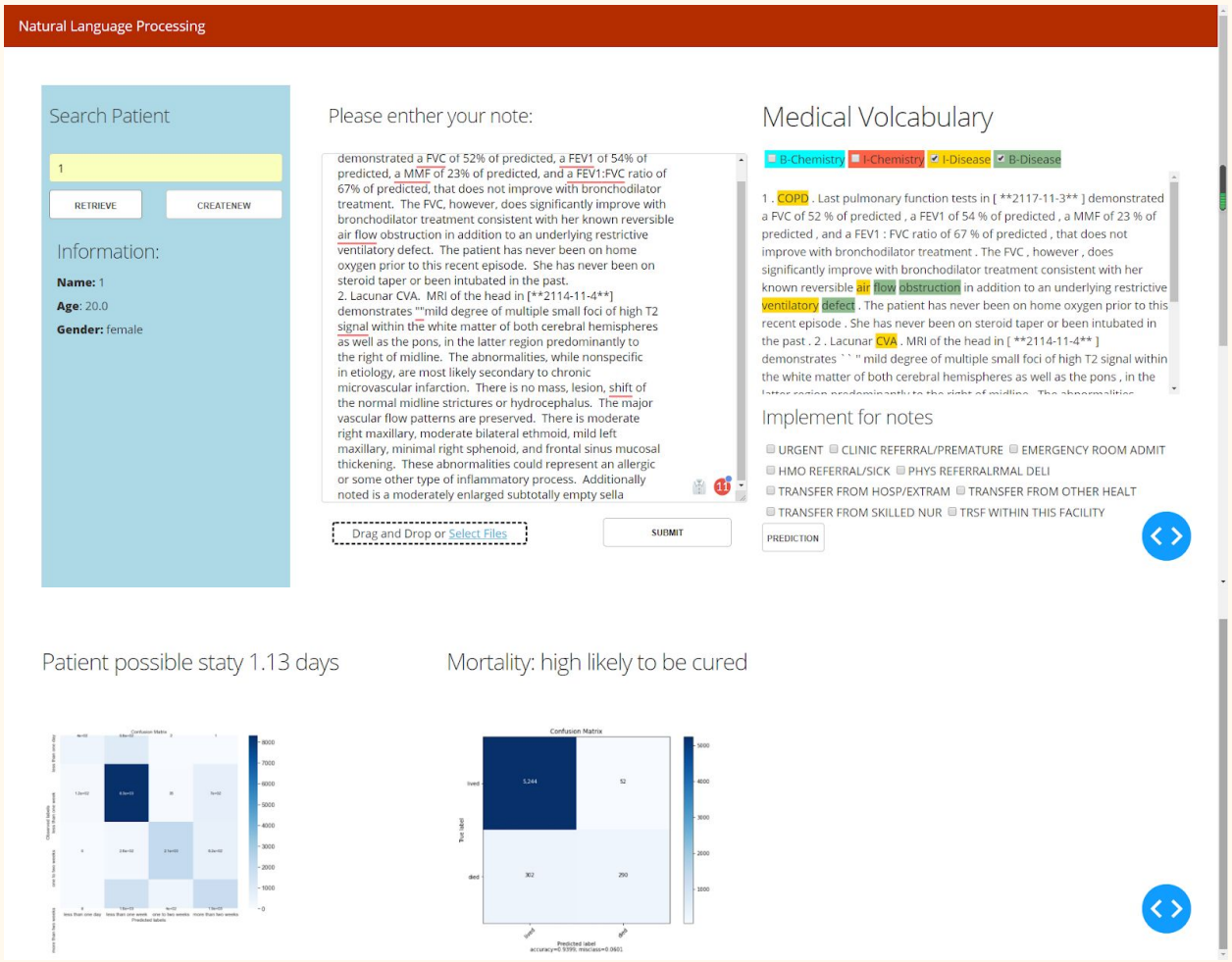
best performance on the internet by far is just around 0.72. Deep learning is already black box itself, and by applying a very complex word vectors make it more like a deeper black box. This indicated the application of pure NLP on clinical notes is still facing a lot of challenges, much harder than what people imagine.

3. Evaluation

The evaluation accuracy is ***0.621760705752619***, which is not quite high. This is because we have an imbalanced dataset. Readmission label 1 vs label 0 is roughly 3 vs 1. Thus we



turn our metrics to the precision-recall curve. As illustrated in the graph, the area under the precision-recall curve is 0.73, which is close to 1.



Data Product

A web application is designed to highlight the name entities and illustrate the models' prediction.

A medical professional can retrieve a patient's historical health condition by providing a unique medical identification, or create a new profile by clicking the 'create new' button and filling the patient's basic personal information.

The new clinical note can be either uploaded in text format or entered by hand. After clicking the 'submit' button, the name entities of each word will be labeled in different colors: cyan for 'B-Chemistry', tomato red for 'I-Chemistry', yellow for 'B-Disease', and green for 'I-Disease'.

Other additional information that have not contained in the text, but may contribute to the prediction model are under the name entities area.

After clicking ‘prediction’. The prediction about patient’s length of stay, mortality, and readmission will be updated spontaneously with models confusion matrix.

For further development. A patient’s interface may also be added to the applications. Which could keeps doctor’s records with explanation on medical vocabulary.

Lessons Learnt

Unstructured data like images, texts are very hard to establish a solid model since the internal variation is huge and unpredictable. The usual challenges of unstructured data are huge dataset and long time experiments required. Here we want to make one more comment on it that unstructured data prediction is also under vicious assault . For example, a spy doctor might write a “looks normal” clinical note but in fact it crushes the model in some way, and we cannot even list the evidence that such a doctor intentionally wrote in that way. Hence, this is the reason why the AI application of security related business is still under development.

Summary

In conclusion, this project collaborates clinical natural language processing with deep learning prediction models. The transformation-ner extracts the medical vocabulary from rare doctor’s notes to parse the notes into features. Then those features will be further processed by machine learning models to make predictions about patients' health conditions.

Reference:

1. Clinical Natural Language Processing with Deep Learning https://link.springer.com/chapter/10.1007/978-3-030-05249-2_5
2. ClinicalBERT: Modeling Clinical Notes and Predicting Hospital Readmission. https://arxiv.org/abs/1904.05342
3. Predicting hospital length-of-stay at time of admission

   https://github.com/daniel-codes/hospital-los-predictor
4. Machine Learning Primer for Clinicians–Part 8

https://histalk2.com/2018/12/12/machine-learning-primer-for-clinicians-part-8/

5. Medical Information Extraction & Analysis: From Zero to Hero with a Bit of SQL and a Real-life Database

   https://www.amazon.com/Medical-Information-Extraction-Analysis-Real-life-ebook/dp/B07BKNVF7C