HOSPITAL READMISSION PREDICTION USING IBM WATSON

1. INTRODUCTION

1.1 Overview

If a hospital has multiple readmissions, it means that the hospital needs to work on the quality of services it is providing with respect to the health and wellness of its patients. Being able to predict whether a person will be readmitted to the hospital within 30 days or not, will be of great help to the hospital in developing an idea of the incoming number of repeated patients which in turn helps to provide better services for patients with increased risk of disease.

One patient population that is at increased risk of hospitalization and readmission is diabetes. Diabetes is a medical condition that affects approximately 1 in 10 patients in the United States. So in this project, we will be focusing on hospital readmission prediction for patients who are having diabetes.

This study used the Health Facts database (Cerner Corporation, Kansas City, MO), a national data warehouse that collects comprehensive clinical records across hospitals throughout the United States. The Health Facts data we used was an extract representing 10 years (1999–2008) of clinical care at 130 hospitals and integrated delivery networks throughout the United States.

1.2 Purpose

The main purpose of this project is to predict whether a person who is suffering from diabetes and consulting a specific hospital will be readmitted or not, based on multiple factors.

We will be using classification algorithms such as Logistic Regression, KNN, Decision tree, Random Forest, AdaBoost, and GradientBoost. We will train and test the data with these algorithms. From this, the best model is selected and saved in pkl format. We will also be deploying our model locally using Flask.

2. LITERATURE SURVEY

2.1 Existing Problem

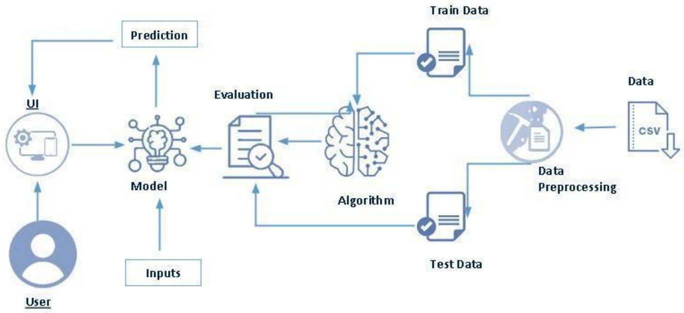
One of the critical problems in healthcare is predicting the likelihood readmission in case of chronic diseases such as diabetes to be able to allocate necessary resources such as beds, rooms, specialists, and medical staff, for an acceptable quality of service. Unfortunately relatively few research studies in the literature attempted to tackle this problem; the majority of the research studies are concerned with predicting the likelihood of the diseases themselves. Numerous machine learning techniques are suitable for prediction. Nevertheless, there is also shortage in adequate comparative studies that specify the most suitable techniques for the prediction process.

2.2 Proposed Solution

This Project presents a comparative study among few common techniques in the literature for predicting the likelihood of hospital readmission in case of diabetic patients. Those techniques are Logistic Regression, KNN, Decision tree, Random Forest, AdaBoost, and GradientBoost... The comparative study is based on realistic data gathered from a number of hospitals in the United States.

3. THEORITICAL ANALYSIS

3.1 Block Diagram



3.2 Hardware / Software Designing

Software

Anaconda Navigator : Anaconda Navigator is a free and open-source distribution of the Python and R programming languages for data science and machine learning-related applications. It can be installed on Windows, Linux, and macOS.Conda is an open-source, cross-platform, package management system. Anaconda comes with so very nice tools like JupyterLab, Jupyter Notebook,QtConsole, Spyder, Glueviz, Orange, Rstudio, Visual Studio Code. For this project, we will be using a Jupyter notebook and Visual Studio Code.

Python Packages

NumPy : NumPy is a Python package that stands for 'Numerical Python. It is the core library for scientific computing, which contains a powerful n-dimensional array of objects.

Pandas : Pandas is a fast, powerful, flexible, and easy-to-use open-source data analysis and manipulation tool, built on top of the Python programming language.

scikit-learn : Scikit-learn is an open source data analysis library, and the gold standard for Machine Learning (ML) in the Python ecosystem.

Matplotlib : Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Matplotlib makes easy things easy and hard things possible. Create publication quality plots. Make interactive figures that can zoom, pan, update.

SciPy : SciPy stands for Scientific Python. It provides more utility functions for optimization, stats and signal processing. Like NumPy, SciPy is open source so we can use it freely. SciPy was created by NumPy's creator Travis Olliphant.

Seaborn : Seaborn is a library that uses Matplotlib underneath to plot graphs. It will be used to visualize random distributions.

Flask : Web framework used for building Web applications.

Hardware

- **Device Name :** LAPTOP-T9PK1LO2
- **Processor :** AMD Ryzen 5 2500U with Radeon Vega Mobile Gfx 2.00 GHz
- **System Type :** 64-bit operating system, x64-based processor

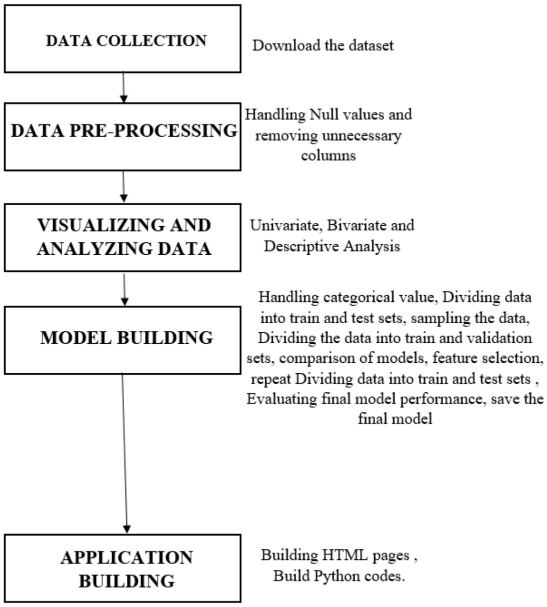
4. EXPERIMENTAL INVESTIGATIONS

There are many popular open sources for collecting the data. Eg: kaggle.com, UCI repository, etc. In this project, we have used diabetic_data.csv. We need to pre-process the collected data before gaining insights and building our model. We need to clean the dataset properly in order to fetch good results. This activity includes handling null values and removing unnecessary columns. Though the dataset seems to be completely free of null values, it is not so. We observe from the head and tail of data that a number of fields are filled with '?'. These are nothing but null values. So in order to get the null values count of each column, replace all '?' in data with np.nan and then find the sum of null values. We use univariate analysis for understanding the data with a single feature. We use bivariate analysis to find the relation between two features. Here we are visualising the relationship of various features with respect to readmitted, which is our target variable. Descriptive analysis is to study the basic statistical features of data. As we can see our dataset has categorical data. Before training our model, we must convert the categorical data into a numeric form. There are multiple encoding techniques to convert the categorical columns into numerical columns. For this project we will be encoding some features manually and some others using OrdinalEncoder(). For splitting the data into train and test sets, we are using the train_test_split() function from sklearn. As parameters, we are passing X, y,stratify, test_size, random_state.

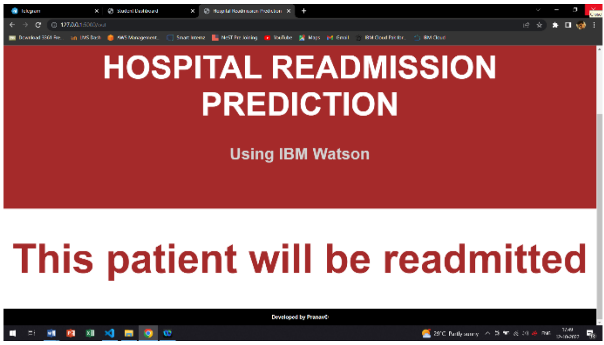
We will be using SMOTE algorithm for oversampling our minority class.

We will be considering multiple models to train our data and choose the one that performs the best.We have trained our model with 29 features. But all these features may not be important for prediction. Hence we will select the features that contribute significantly to the model performance. We will compare the confusion matrix, ROC curve and classification report for both models. In order to obtain these, we will be using the confusion_matrix(),roc_curve() and classification_report() functions from sklearn.metrics. The final step is saving our model. We can do it by using pickle.dump(). After the model is build, we integrate it to the web application.

5. FLOWCHART


DATA COLLECTION
Download the dataset
DATA PRE-PROCESSING
Handling Null values and removing unnecessary columns
VISUALIZING AND ANALYZING DATA
Univariate, Bivariate and Descriptive Analysis
MODEL BUILDING
Handling categorical value, Dividing data into train and test sets, sampling the data, Dividing the data into train and validation sets, comparison of models, feature selection, repeat Dividing data into train and test sets , Evaluating final model performance, save the final model
APPLICATION BUILDING
Building HTML pages , Build Python codes.

6. RESULT

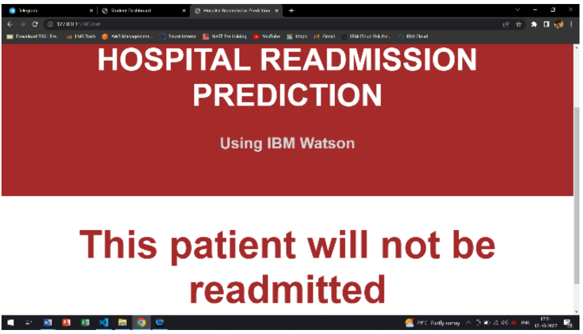


HOSPITAL READMISSION PREDICTION

Using IBM Watson

This patient will be readmitted

Developed by Pranav©



HOSPITAL READMISSION PREDICTION

Using IBM Watson

This patient will not be readmitted

The Project will predict whether the patient will readmit to hospital or not.

7. ADVANTAGES AND DISADVANTAGES

Advantages

- Helps to provide better quality of services to patients.
- Easy to access
- Easy to predict whether a patient will be readmitted to the hospital within 30 days or not.

- Predicting the likelihood readmission in case of chronic diseases such as diabetes to be able to allocate necessary resources such as beads, rooms, specialists and medical staffs.

Disadvantage

- May not be that much familiar for normal people.
- Initial Implementation cost

8. APPLICATIONS

- Hospital Sector
- Health care
- Diabetics Treatment

9. CONCLUSIONS

The main purpose of this project is to predict whether a person who is suffering from diabetes and consulting a specific hospital will be readmitted or not, based on multiple factors. This Project presents a comparative study among few common techniques in the literature for predicting the likelihood of hospital readmission in case of diabetic patients. Those techniques are Logistic Regression, KNN, Decision tree, Random Forest, AdaBoost, and GradientBoost... The comparative study is based on realistic data gathered from a number of hospitals in the United States.

10. FUTURE SCOPE

One patient population that is at increased risk of hospitalization and readmission is diabetes. Diabetes is a medical condition that affects approximately 1 in 10 patients in the United States. So in this project, we will be focusing on hospital readmission prediction for patients who are having diabetes.

11.BIBLIOGRAPHY

- **Application of Machine Learning in Hospital Readmission Prediction**
  https://bmcmedresmethodol.biomedcentral.com/articles/10.1186/s12874-021-01284-z
- **https://www.w3schools.com/css/css_rwd_templates.asp**
- **https://smartinternz.com/**

APPENDIX

app.py

```
from flask import Flask, render_template, request
import pickle, joblib
import pandas as pd
app = Flask(__name__)
model = pickle.load(open("model.pkl","rb"))
ct = joblib.load('feature_values')
@app.route('/')
def home():
    return render_template("home.html")
@app.route('/pred')
def predict():
    return render_template("index.html")
@app.route('/out', methods =["POST"])
def output():
    discharge_disposition_id = request.form["discharge_disposition_id"]
    admission_source_id = request.form["admission_source_id"]
    time_in_hospital = request.form["time_in_hospital"]
    num_medications = request.form["num_medications"]
    number_emergency = request.form["number_emergency"]
    number_inpatient = request.form["number_inpatient"]
    diag_1 = request.form["diag_1"]
    diag_2 = request.form["diag_2"]
    max_glu_serum = request.form["max_glu_serum"]
    glimepiride = request.form["glimepiride"]
    diabetesMed = request.form["diabetesMed"]
    if max_glu_serum == '>200' or max_glu_serum=='>300':
        max_glu_serum=1
    elif max_glu_serum=='Norm':
        max_glu_serum=0
    else:
        max_glu_serum=-99

    if glimepiride == 'No':
        glimepiride = 0
    else:
        glimepiride=1
    data = [[discharge_disposition_id,admission_source_id,time_in_hospital,
            num_medications, number_emergency, number_inpatient,diag_1, diag_2,
            max_glu_serum, glimepiride, diabetesMed]]
    feature_cols = ['discharge_disposition_id', 'admission_source_id', 'time_in_hospital',
                    'num_medications','number_emergency', 'number_inpatient',
                    'diag_1', 'diag_2','max_glu_serum', 'glimepiride', 'diabetesMed']

    pred = model.predict(ct.transform(pd.DataFrame(data,columns=feature_cols)))
    pred = pred[0]
    if pred:
        return render_template("output.html",y="This patient will be readmitted ")
    else:
        return render_template("output.html",y="This patient will not be readmitted")
if __name__ == '__main__':
    app.run(debug = True)
```

home.html

```
<html>
 <head>
  <title>HRP</title>
  <link href="../static/css/mainstyle.css" rel="stylesheet" type="text/css">
 </head>
 <body>
  <div class="nav">
    <a href="/">Home</a>
    <a href="/pred">Prediction</a>
  </div>
  <div class="header">
   <h1>Hospital Readmission Prediction</h1>
   <h2>Using IBM Watson</h2>
  </div>
<br>
  <div class="row">
   <div class="column">
    <div class="card">
     <div class="icon-wrapper">
      <i class="fas fa-hammer"></i>
     </div>
     <h3>Objective</h3>
     <p>
      The prime objective of this project is to predict whether a person who has been diagnosed with diabetes will
be readmitted to the hospital within 30 days or not.This will help the hospitals in being ready for any number of
patients that could be readmitted.
     </p>
    </div>
   </div>
   <div class="column">
    <div class="card">
     <div class="icon-wrapper">
      <i class="fas fa-brush"></i>
     </div>
     <h3>Method</h3>
     <p>
      The problem associated is binary classification.  Since the target variable in the dataset is quite imbalanced,
SMOTE algorithm was used to oversample the minority class.
     </p>
    </div>
   </div>
   <div class="column">
    <div class="card">
     <div class="icon-wrapper">
      <i class="fas fa-wrench"></i>
     </div>
     <h3>Models</h3>
     <p>
      A number of models were used such as logistic regression, KNN Classifier, Decision Tree Classifier,
Random Forest Classifier, Adaboost Classifier etc.Out of all the models, random forest classifier is found to perform
the best.
     </p>
    </div>
   </div>
```

```
      </div>
    </div>
<br>
<footer class="footer">
  <h1>Developed by Pranav&copy;</h1>
</footer>
  </body>
</html>
```

index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Hospital Readmission Prediction</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="../static/css/mainstyle.css" rel="stylesheet" type="text/css">
</head>
<body>
  <div class="nav">
    <a href="/">Home</a>
    <a href="/pred">Prediction</a>
</div>
<div class="header">
  <h1>HOSPITAL READMISSION PREDICTION</h1>
  <h2>Using IBM Watson</h2>
</div>
<div class="row">
  <div class="main2">
    <form  action = "/out" method = "post" >
    <p>
        Discharge description
        <select name= "discharge_disposition_id">
        <option> Select option </option>
        <option value = 1> Discharged to home </option>
        <option value = 2> Discharged to another facility for inpatient care </option>
        <option value = 10> Discharged to another facility for outpatient services </option>
        <option value = 18> Unknown </option>
        </select>
        </p>
        <p>
        Admission description
            <select name= "admission_source_id">
            <option> Select option </option>
            <option value = 1> Referral </option>
            <option value = 4> Transfer </option>
            <option value = 9> Unknown </option>
            <option value = 11> Just born </option>
            </select>
            </p>
   <!-- <label for="appointment_message" class="text-black">Message</label> -->
            <p> Time in Hospital
           <input class="form1" type = "number" name= "time_in_hospital"/> </p>
            <!-- <label for="appointment_message" class="text-black">Message</label> -->
            <p> Number of Medications
             <input  class="form1"  type = "number" name= "num_medications"/> </p>
            <!-- <label for="appointment_message" class="text-black">Message</label> -->
            <p> Number of emergencies
             <input  class="form1"  type = "number" name= "number_emergency"/> </p>
            <!-- <label for="appointment_message" class="text-black">Message</label> -->
            <p> Number of inpatient visits
            <input  class="form1" type = "number" name= "number_inpatient"/> </p>
            <p>
              Primary Diagnosis
              <select name= "diag_1">
              <option> Select option </option>
              <option value = "circulatory"> Circulatory </option>
              <option value = "digestive"> Digestive </option>
              <option value = "genitourinary"> Genitourinary </option>
              <option value = "diabetes"> Diabetes </option>
              <option value = "injury"> Injury </option>
              <option value = "musculoskeletal"> Musculoskeletal </option>
              <option value = "neoplasms"> Neoplasms </option>
              <option value = "respiratory"> Respiratory </option>
              <option value = "other"> Other </option>
              </select>
              </p>
              <p>
                Secondary Diagnosis
                 <select name= "diag_2">
                 <option> Select option </option>
                 <option value = "circulatory"> Circulatory </option>
                 <option value = "digestive"> Digestive </option>
                 <option value = "genitourinary"> Genitourinary </option>
                 <option value = "diabetes"> Diabetes </option>
                 <option value = "injury"> Injury </option>
                 <option value = "musculoskeletal"> Musculoskeletal </option>
                 <option value = "neoplasms"> Neoplasms </option>
                 <option value = "respiratory"> Respiratory </option>
                 <option value = "other"> Other </option>
                 </select>
                 </p>
                 <p>
                   Glucose serum test result
                   <select name= "max_glu_serum">
                   <option> Select option </option>
                   <option value = "None"> Not measured </option>
                   <option value = "Norm"> Normal </option>
                   <option value = ">200"> Greater than 200 </option>
                   <option value = ">300"> Greater than 300 </option>
                   </select>
                 </p>
                 <p>
                   Glimepiride dosage
             <select name= "glimepiride">
             <option> Select option </option>
             <option value = "Up"> Dosage increased </option>
             <option value = "Down"> Dosage decreased </option>
             <option value = "Steady"> Dosage did not change </option>
             <option value = "No"> Dosage not prescribed </option>
             </select>
             </p>
        <p>
              Diabetes medication prescribed?
```

```
                <select name= "diabetesMed">
                <option> Select Yes or No </option>
                <option value = "Yes"> Yes </option>
                <option value = "No"> No </option>
                </select>
                </p>
                <input type="submit" class="btna" name="s" value="Predict"><br><br>
            </form>
            <a href="/"><button class="btnb">Home</button></a>
    </div>
</div>
<footer class="footer">
  <h1>Developed by Pranav&copy;</h1>
</footer>
</body>
</html>
```

output.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Hospital Readmission Prediction</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="../static/css/mainstyle.css" rel="stylesheet" type="text/css">
</head>
<body>
  <div class="nav">
    <a href="/">Home</a>
    <a href="/pred">Prediction</a>
</div>
<div class="header">
  <h1>HOSPITAL READMISSION PREDICTION</h1>
  <h2>Using IBM Watson</h2>
</div>

<div class="row3">
<h1>{{y}}</h1>
</div>
<footer class="footer">
    <h1>Developed by Pranav&copy;</h1>
  </footer>
</body>
</html>
```