

A REVIEW OF LIVER PATIENT ANALYSIS USING MACHINE LEARNING

TEAM LEADER

S. AASIKA

TEAM MEMBERS

A. ANITHA

A. AMALA JEFRIYA

M. DHARSHINI

S. VINOOTHINI

CONTENTS

INTRODUCTION

PROBLEM DEFINITION & DESIGN THINKING

RESULT

ADVANTAGES AND DISADVANTAGES

APPLICATIONS

CONCLUSION

FUTURE SCOPE

APPENDIX

INTRODUCTION

Liver diseases averts the normal function of the liver. This disease is caused by an assortment of elements that harm the liver. Diagnosis of liver infection at the preliminary stage is important for better treatment. In today's scenario devices like sensors are used for detection of infections. Accurate classification techniques are required for automatic identification of disease samples. This disease diagnosis is very costly and complicated. Therefore, the goal of this work is to evaluate the performance of different Machine Learning algorithms in order to reduce the high cost of liver disease diagnosis. Early prediction of liver disease using classification algorithms is an efficacious task that can help the doctors to diagnose the disease within a short duration of time. In this project we will analyse the parameters of various classification algorithms and compare their predictive accuracies so as to find out the best classifier for determining the liver disease. This project compares various classification algorithms such as Random Forest, Logistic Regression, KNN and ANN Algorithm with an aim to identify the best technique. Based on this study, Random Forest with the highest accuracy outperformed the other algorithms and can be further utilised in the prediction of liver disease and can be recommended to the user.

PROBLEM DEFINITION AND DESIGN THINKING

i) Specify The Business Problem

Liver diseases averts the normal function of the liver. This disease is caused by an assortment of elements that harm the liver. Diagnosis of liver infection at the preliminary stage is important for better treatment. In today's scenario devices like sensors are used for detection of infections. Accurate classification techniques are required for automatic identification of disease samples. This disease diagnosis is very costly and complicated. Therefore, the goal of this work is to evaluate the performance of different Machine Learning algorithms in order to reduce the high cost of liver disease diagnosis. Early prediction of liver disease using classification algorithms is an efficacious task that can help the doctors to diagnose the disease within a short duration of time. In this project we will analyse the parameters of various classification algorithms and compare their predictive accuracies so as to find out the best classifier for determining the liver disease. This project compares various classification algorithms such as Random Forest, Logistic Regression, KNN and ANN Algorithm with an aim to identify the best technique. Based on this study, Random Forest with the highest accuracy outperformed the other algorithms and can be further utilised in the prediction of liver disease and can be recommended to the user.

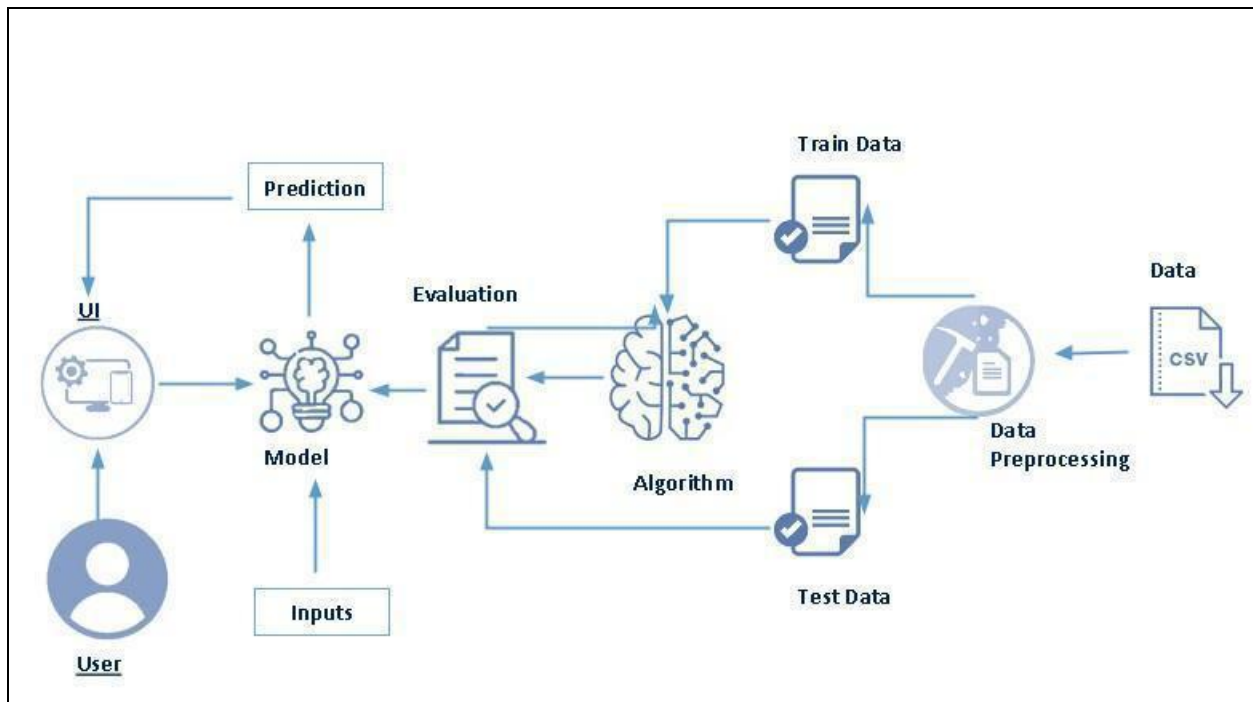
ii) Business Requirements

Patients with Liver disease have been continuously increasing because of excessive consumption of alcohol, inhale of harmful gases, intake of contaminated food, pickles and drugs and other factors. This dataset was used to evaluate prediction algorithms in an effort to reduce burden on doctors. Use these patient records to build a prediction model that will predict which patients have liver disease and which ones do not.

iii) Social Or Business Impact

Social Impact:- Today almost everybody above the age of 12 years has smartphones with them, and so we can incorporate these solutions into an android app or ios app. Also it can be incorporated into a website and these app and website will be highly beneficial for a large section of society.

Business Model/Impact:- Its now more feasible Blood test centers to give the result. As for this model user don't need to have any deep knowledge of medical science and liver diseases. User need to do pass the details being asked, which are already present in the blood test report(some like age, gender are already known) and then user will get the results of prediction.



RESULT

- Open Command prompt from the start menu
- Navigate to the folder where your python script is.
- Now type “python app.py” command
- Navigate to the localhost where you can view your web page.
- Click on the predict button from the top left corner, enter the inputs, click on the submit button, and see the result/prediction on the web

LIVER DISEASE PREDICTION

AGE
Total_Bilirubin
Direct_Bilirubin
Alkaline_Phosphatase
Alamine_Aminotransferase
Aspartate_Aminotransferase
SEC
Albumin
Albumin_and_Globulin_Rat
gender
<input type="button" value="Predict"/>
{{ prediction }}

ADVANTAGES & DISADVANTAGES

ADVANTAGES

- Liver Transplant
- Diagnosis, grades and stages:
 1. Hepatitis C
 2. Hepatitis B
 3. Steatohepatitis
 4. Autoimmune hepatitis
- Identifies hepatotoxicity
- Evaluates abnormal liver function tests

DISADVANTAGES

- Invasive
- Accessibility to the procedure

- Need for training
- Repeated testing
- Cost

APPLICATIONS

Planning phase include the creation of ideas to support healthcare and technical team through the prediction of liver diseases. The main objective of planning phase is to plan the step involved in the development of prediction system using software engineering life cycle. In addition, challenging think is to remove the gap between the software development members and health-care specialist. In the analysis phase, the concern is to gather prediction system requirements and environmental considerations. The requirements involve the people from a different background area such as informaticists, physicians, patients etc.

CONCLUSION

In the proposed work, different classifiers were implemented on liver patient diseases dataset to predict liver diseases based on developed software. Dataset was processed and implemented on WEKA tool using feature selection techniques with 10-fold cross validation testing option. The results of the proposed work were compared using feature selection and without using feature selection techniques after the implementation of different classifiers in terms of execution time and accuracy. During the research work the result of other parameters like kappa statistic, correctly classified instances, and mean absolute error were also compared on liver patient diseases dataset. The best result was achieved using Logistic Regression classifier with feature selection techniques and execution time of different classifiers was decreased after the implementation of feature selection technique

FUTURE SCOPE

Liver diseases have produced a big data such as metabolomics analysis, electronic health records, and report including patient medical information and disorders. However, these data must analysed and integrated if they are produced models about physiological mechanisms of pathogenesis.

APPENDIX

```

2         import numpy as np
3         import pandas as pd
4         from flask import Flask, request, render_template

```

```

5         import pickle
6
7         app = Flask(__name__)
8         model = pickle.load(open('liver1.pk', 'rb'))
9
10        @app.route('/')
11        def home():
12            return render_template('index.html')
13
14        @app.route('/predict',methods=['POST'])
15        def predict():
16            '''
17            For rendering results on HTML GUI
18            '''
19            int_features = [x for x in request.form.values()]
20
21            df = pd.DataFrame([int_features])
22
23            df1=df.replace(["male","female"], [1,0])
24
25
26
27            final_features = np.array(df1)
28
29            final_features = final_features.reshape(1,10)
30            # final_features = pd.DataFrame([final_features])
31
32            if model.predict(final_features) ==[1]:
33                predict = "you have DISEASES related to liver"
34            else:
35                predict = "you dont have DISEASES related to liver"
36
37
38
39
40
41
42
43

```

```
44         return render_template('index.html',prediction=predict)
45
46
47     if __name__ == "__main__":
48         app.run(debug=True)
```