Revolutionizing Liver Care: Predicting Liver Cirrhosis using Advanced Machine Learning Techniques

Project Description

Liver cirrhosis is a major global health concern, characterized by permanent liver damage and scarring (fibrosis), which impairs the liver's ability to function effectively. This chronic and progressive disease is often diagnosed at a late stage when liver function has already deteriorated significantly. Early detection is critical for improving prognosis and increasing the effectiveness of treatments.

This project proposes the use of advanced Machine Learning (ML) techniques to predict the risk of liver cirrhosis based on a set of clinical features. By developing a predictive model and integrating it into a user-friendly Flask web application, we aim to provide healthcare professionals and researchers with a decision-support tool that offers accurate, real-time predictions. This system can help prioritize patient cases, recommend early interventions, and reduce overall healthcare costs.

Problem Description

Medical Context

Liver cirrhosis occurs due to long-term liver damage caused by conditions like hepatitis, chronic alcoholism, or fatty liver disease. Over time, healthy liver tissue is replaced by scar tissue, leading to loss of function. Without timely intervention, cirrhosis can progress to liver cancer or liver failure.

Current Challenges

Diagnosis often happens late in the disease cycle

Manual diagnosis can be time-consuming and error-prone

Requires specialized testing that may not be readily available in all settings

Problem Statement

Can machine learning algorithms accurately predict liver cirrhosis using non-invasive patient data, and can we deploy this prediction in a real-time web application to support clinical decision-making?

Literature Survey

Numerous studies have demonstrated the usefulness of ML in liver disease detection:

Supervised ML techniques such as Decision Trees, Random Forest, and XGBoost have shown promise in healthcare classification tasks.

Tools like Liver Cirrhosis Diagnosis Support Systems (LCDSS) were proposed for early-stage detection using statistical modeling.

Despite promising results, many systems lack web integration or require domain expertise to use effectively.

Gaps Identified

Most systems are offline and not user-friendly for real-time clinical use.

Many models are not optimized or interpreted in ways understandable to non-technical healthcare professionals.

Objective of This Project

Build an accurate, scalable ML model for liver cirrhosis prediction.

Develop a real-time, accessible web-based prediction interface.

Improve health outcomes through early detection and better resource allocation.

Machine Learning Algorithms Used

Decision Tree Classifier – Simple, interpretable tree-based model.

Random Forest Classifier – Ensemble of decision trees for better generalization.

K-Nearest Neighbors (KNN) – Distance-based method for classification.

Evaluation Metrics

Accuracy – Overall correctness of the model.

Precision – How many predicted positives were truly positive.

Recall – How many actual positives were correctly predicted.

F1-Score – Harmonic mean of precision and recall.

ROC-AUC Score – How well the model distinguishes between classes.

Tech Stack

Python – Core language for scripting and modeling

Pandas, NumPy – Data preprocessing and analysis

Scikit-learn – Model building and evaluation

Matplotlib, Seaborn – Exploratory Data Analysis (EDA)

Flask – Backend web framework for deployment

HTML/CSS/JS – User interface design

Project Workflow

1. Problem Understanding – Defined the use case and real-world need

2. Data Collection – Used liver patient dataset from UCI or verified medical source

3. Data Preprocessing – Removed missing values, normalized data

4. EDA (Exploratory Data Analysis) – Analyzed feature distributions and correlations

5. Model Building – Trained Decision Tree, Random Forest, KNN, and XGBoost models

6. Hyperparameter Tuning – Improved accuracy with GridSearchCV and manual tuning

7. Model Saving – Saved the best model and normalizer using pickle

8. Flask Integration – Developed a web app for input and real-time prediction

Project Folder Structure

```
LIVER_CIRRHOSES/
├── data/                    # Raw or preprocessed datasets
├── documentation/           # Project report, references, and additional docs
├── flask/
│   ├── env/                 # Virtual environment (optional)
│   ├── static/              # CSS, JS, and image files for styling
│   └── templates/           # HTML templates for Flask
│       ├── index.html       # Home page or input form
│       ├── liver.html       # Intermediate or result routing page
│       └── result.html      # Output prediction result display
├── app.py                   # Main Flask backend application
├── training/
│   ├── Liver_Disease_prediction (1).ipynb  # Model development notebook
│   └── LiverDisease.pkl                    # Saved machine learning model
```

References

1. Supervised Learning – https://www.javatpoint.com/supervised-machine-learning

2. Random Forest – https://www.javatpoint.com/machine-learning-random-forest-algorithm

3. KNN – https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning

4. Evaluation Metrics
–https://www.analyticsvidhya.com/blog/2019/08/11-important-model-evaluation-error-metrics/

5. Flask Basics – https://www.youtube.com/watch?v=lj4I_CvBnt0

How to Run the Project

1. Clone the Repository

git clone

2. Install Required Libraries

pip install -r requirements.txt

3. Run the Flask App

python app.py

4. Open in Browser

Go to: http://localhost:5000