Interpretable Liver Disease Prediction System Using XGBoost with SHAP Analysis

Shamshadh K V M*, Mrs. Geetha N B**
*(Computer Science & Engineering, University BDT College, Davanagere
Email: Shamshadhkvm@gmail.com)
**(Computer Science & Engineering, University BDT College, Davanagere
Email: geethanb291979@gmail.com)

----------**********----------

Abstract:

Liver illness is among the major health problems throughout the world and is often detected only in later stages, which makes treatment difficult. Early identification is important, However, conventional approaches have drawbacks as the symptoms are not always clear. In this project, a machine learning-based approach is used to predict liver disease using the XGBoost algorithm. The dataset contains demographic and biochemical details related to liver function. However, conventional approaches have drawbacks using a hybrid k-Nearest Neighbour (KNN) imputation approach, and the data was normalized before training. The model's performance was evaluated using accuracy, sensitivity, specificity, and confusion matrix. The XGBoost model gave an accuracy of 92.4%, which is better compared to Logistic Regression. To make the model more interpretable, SHAP values were used to determine the most important features influencing the prediction. Finally, the prototype was deployed in a Django web application with features like secure login, prediction form, history management, and storage of user data. This experiment demonstrates how machine learning combined with web technology can provide a helpful instrument for anticipating prediction of liver disease.

Keywords— **Liver Disease Prediction, XGBoost, SHAP Analysis, Machine Learning, Interpretability, Django Web Application.**

----------**********----------

1. INTRODUCTION

Liver diseases are a major worldwide concern, impacting millions of people across different regions and communities. These conditions include hepatitis, fatty liver disease, cirrhosis, and hepatocellular carcinoma. A key difficulty in managing liver diseases lies in their subtle or non-specific early symptoms, which often delay diagnosis and consequently lead to poor patient outcomes. This underscores the growing need for cost-effective and non-invasive screening techniques that aid in clinicians in early detection and treatment planning. Recent progress in machine learning (ML) and artificial intelligence (AI) has reshaped medical diagnostics by uncovering hidden patterns in patient data and generating precise prediction outcomes. In this work, a predictive system is built around the eXtreme Gradient Boosting (XGBoost) algorithm, a method known for its speed, strong regularization, and superior classification accuracy. The model is developed to distinguish liver disease conditions using both clinical indicators and demographic attributes. The methodology incorporates advanced preprocessing steps, hyperparameter tuning for model optimization, and explainability through SHAP (SHapley Additive exPlanations) analysis, ensuring both transparency and soundness of predictions. Furthermore, the trained model is integrated

to a Django-based web platform that offers an interactive interface for users. This platform enables secure input of test data, real-time predictions, review of historical records, and

account management, thereby bridging machine learning solutions with practical healthcare applications.

2. LITERATURE REVIEW

Many scholars have investigated machine learning techniques to forecast and diagnose liver diseases by utilizing the Indian Liver Patient Dataset (ILPD) and other related clinical records. Early work by Rajeswari and Reena [1] applied Naive Bayes and Decision Tree classifiers to the ILPD, achieving close to 70% accuracy but showing poor specificity, which indicated that such basic models were unsuitable for dependable clinical use. Expanding on this work, Singh and Kaur [2] examined many classifiers, such as Support Vector Machines (SVM), Random forest, and k-Nearest Neighbors (k-NN). Their study's conclusions showed that while SVM achieved strong sensitivity, its specificity was not so strong, particularly when handling with the issue of imbalanced datasets—a frequent challenge in medical research. Kalra et al. [3] implemented Logistic Regression and Random Forest techniques for liver disease prediction. While Random Forest achieved around 85% accuracy, its lack of interpretability limited its acceptance in clinical decision-making, where transparency of the model's reasoning is crucial. To address generalization challenges, Choubey and Paul [4] proposed an ensemble-based model that combined Gradient Boosting and AdaBoost algorithms. Their findings demonstrated improved predictive performance when multiple classifiers were integrated, though the study did not apply modern explainability tools such as SHAP to make the predictions more transparent for healthcare professionals. More recently, Vaidya and Patil [5] applied the XGBoost algorithm on a preprocessed liver dataset and reported higher accuracy compared to conventional classifiers. However, the absence of real-time deployment or web-based implementation restricted the practical impact of their work, indicating that further research is still needed in developing deployable systems.

3. METHODOLOGY

3.1 System Implementation

The proposed Interpretable Liver Disease Prediction System integrates advanced machine learning techniques with a secure, web-based interface to assist in clinical decision-making. The system enables healthcare professionals or users to input patient parameters, obtain real-time predictions on liver disease risk, and manage previous prediction records in an organized and confidential manner.

The implementation consists of two major components:

3.1.1 Machine Learning Component

The predictive backbone of the system is developed using the **XGBoost (Extreme Gradient Boosting)** algorithm, well-known for its robustness and superior performance in classification tasks.

- Before training, missing values were treated using a **hybrid K-Nearest Neighbour (KNN) imputation** method, ensuring data completeness without compromising data quality.
- All numerical features were normalized to maintain uniform scale and enhance convergence during model training.
- The trained model was evaluated through multiple performance metrics—**accuracy, sensitivity, specificity, precision, F1-score,** and **confusion matrix**—to ensure its reliability.
- To improve interpretability, **SHAP (SHapley Additive exPlanations)** analysis was employed to identify and rank the features contributing most to prediction outcomes.

3.1.2 Web Application Component

The finalized XGBoost model was integrated into a **Django-based web application**, providing a simple yet secure interface for users.

- The platform allows users to **sign up, log in,** and securely interact with the prediction module.

- A **data entry form** facilitates input of clinical parameters, triggering the model to generate and display prediction results instantly.
- The application includes **history management features**, allowing users to view or delete prior prediction records while ensuring **data privacy** through controlled access and secure storage mechanisms.

3.2 Dataset Preparation

3.2.1 Source of Dataset

The study utilized a clinical liver disorder dataset comprising 2,500 patient records. Each record includes demographic and biochemical attributes crucial for liver function assessment, such as Age, Gender, Total Bilirubin, Direct Bilirubin, Alkaline Phosphatase (ALP), Alanine Aminotransferase (ALT), Aspartate Aminotransferase (AST), Total Proteins, Albumin, and Albumin-to-Globulin Ratio.

3.2.2 Handling Missing Values

Approximately 10% of the dataset contained missing entries. Instead of discarding incomplete records, a hybrid KNN-based imputation approach was applied. This technique identifies the most similar records and estimates the missing values using feature averages, thereby preserving the natural relationships between variables and maintaining dataset integrity.

3.2.3 Feature Selection and Encoding

To ensure compatibility with the XGBoost algorithm, categorical attributes were converted into numeric format. The Gender feature was encoded as *0 for Female* and *1 for Male*. All clinical features were retained, as each had direct clinical relevance to liver health prediction, avoiding unnecessary feature elimination.

3.2.4 Data Splitting

To validate the model's predictive performance, the dataset was partitioned into:

- **Training Set (80%)** – used for learning model parameters.
- **Testing Set (20%)** – used for performance evaluation on unseen data.

3.3 Pre-Processing Pipeline

The preprocessing phase transformed raw clinical data into a clean and consistent format suitable for machine learning. The following steps were systematically applied:

1. **Handling Missing Values** – A hybrid KNN-based imputation strategy filled missing entries using the mean of the closest feature vectors, preserving key statistical patterns.
2. **Feature Scaling / Normalization** – Numerical attributes with differing ranges (e.g., enzyme levels and protein concentrations) were normalized to ensure equal contribution during model training.
3. **Encoding Categorical Variables** – The categorical *Gender* variable was numerically encoded to make it compatible with the XGBoost model

3.4 Model Construction

The predictive framework was constructed using the **XGBoost** algorithm, a scalable and efficient implementation of gradient boosting that builds an ensemble of decision trees.

3.4.1 Working Principle of XGBoost

XGBoost constructs trees sequentially, where each new tree corrects the residual errors of the previous one.

- The process starts with a weak learner that makes initial predictions.
- Residuals (errors) are calculated as the difference between predicted and actual outcomes.
- Subsequent trees are trained to minimize these residuals using gradient descent.
- A **regularization term** is incorporated into the objective function to prevent overfitting and control model complexity.

- The final prediction is produced by aggregating the weighted outputs of all trees.

This ensemble approach allows XGBoost to achieve high predictive accuracy and handle missing data efficiently.

3.4.2 Model Configuration and Application

Key hyperparameters such as learning rate, maximum tree depth, and number of estimators were fine-tuned using GridSearchCV. The preprocessed dataset was supplied as input for training, and the resulting optimized model was serialized and saved as best_xgb_model.pkl for integration into the Django web framework.

3.5 Training Procedure

The training phase transformed pre-processed data into a functional predictive model:

1. **Input Data Feeding:** The 80% training data containing demographic and biochemical attributes was provided to the XGBoost classifier.
2. **Sequential Tree Building:** Each tree minimized the residuals from its predecessors, progressively enhancing prediction accuracy.
3. **Optimization via Gradient Descent:** The loss function was iteratively minimized by updating model parameters in the direction of the steepest descent.
4. **Regularization:** Hyperparameters controlling model complexity were fine-tuned to prevent overfitting and enhance generalization.

Once training stabilized, the final model was saved for deployment in the web interface, enabling real-time predictions.

3.6 Performance Evaluation

The trained model was evaluated using the 20% testing subset to verify its predictive capability and reliability.

3.6.1 Evaluation Metrics

The following metrics were computed:

- **Accuracy:** Overall proportion of correct classifications.
- **Sensitivity (Recall):** Ability to correctly detect patients with liver disease.
- **Specificity:** Ability to correctly identify healthy individuals.
- **Precision:** Proportion of true positives among predicted positives.
- **F1-Score:** Harmonic mean of precision and recall, providing a balanced measure of model performance.

All metrics were derived from the confusion matrix, which summarizes true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).

3.6.2 Confusion Matrix and ROC-AUC Analysis

The confusion matrix helped assess classification balance between diseased and healthy cases. The Receiver Operating Characteristic (ROC) curve was plotted to evaluate discriminative power, and the Area Under Curve (AUC = 0.9296) confirmed the high accuracy and reliability of the XGBoost classifier.

3.6.3 SHAP-Based Feature Interpretability

To enhance transparency, SHAP analysis was employed to interpret the influence of each feature on model predictions. The analysis revealed that Direct Bilirubin, Total Bilirubin, Aspartate Aminotransferase (AST), and Alanine Aminotransferase (ALT) were the most influential predictors. Features such as Age, Albumin, and Albumin-to-Globulin Ratio also contributed significantly, aligning with medical understanding of liver health indicators. This interpretability ensures that the model's predictions are both clinically relevant and trustworthy.

4. MATERIALS AND METHODS

The proposed liver disease prediction framework was developed through a structured workflow consisting of dataset collection, preprocessing, model training, and deployment. The Indian Liver Patient Dataset (ILPD) [11], containing ten clinical

attributes such as age, gender, bilirubin concentration, liver enzyme levels, protein values, and the albumin-to-globulin ratio, was utilized for this study.

To ensure data quality and consistency, preprocessing steps were applied, including the imputation of missing values, encoding of categorical variables, and feature normalization. For classification, the Extreme Gradient Boosting (XGBoost) algorithm [6] was selected due to its strong accuracy and robustness in handling imbalanced datasets.

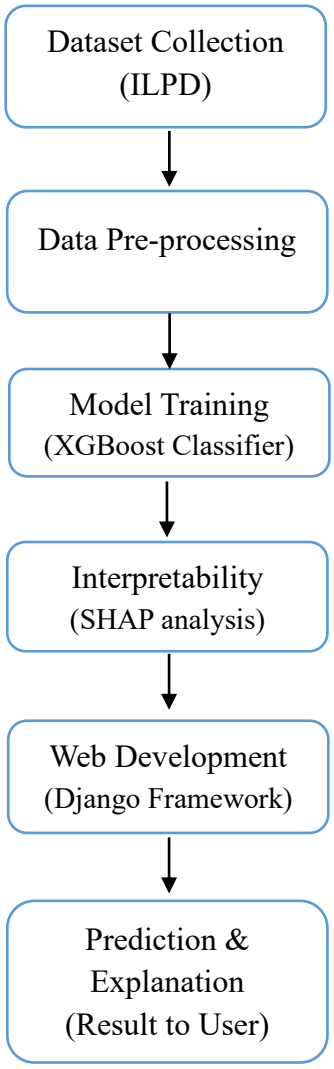

Figure: Flowchart of the System

5. RESULTS AND DISCUSSION

The performance of the proposed system was evaluated using the Indian Liver Patient Dataset (ILPD) [11]. After preprocessing steps such as feature normalization, missing value imputation, and categorical variable encoding, multiple machine learning algorithms were implemented and compared to assess their effectiveness in predicting liver disease. Model performance was evaluated through cross-validation and test accuracy, along with additional classification metrics including sensitivity, specificity, precision, recall, and F1-score.

Table 1 summarizes the comparative performance of the examined models. Among them, the Extreme Gradient Boosting (XGBoost) classifier achieved the highest test accuracy of 92.4%, marginally outperforming Random Forest (92.2%) and demonstrating clear superiority over Gradient Boosting (86.6%), Decision Tree (85.4%), and K-Nearest Neighbors (83.2%). In contrast, Logistic Regression, Support Vector Machine, and Naïve Bayes yielded considerably lower accuracy scores, underscoring their limited effectiveness on this dataset.

Table 1: Model Comparison (Cross-Validation and Test Accuracy)

Model	CV Accuracy (Mean)	Test Accuracy
XGBoost	0.9160	0.924
Random Forest	0.9170	0.922
Gradient Boosting	0.8570	0.866
Decision Tree	0.8565	0.854
K-Nearest Neighbors	0.8195	0.832
Logistic Regression	0.7120	0.712
Support Vector Machine	0.7140	0.708
Naive Bayes	0.5460	0.556

Table 2 presents the comparative results of sensitivity and specificity across the evaluated models. The XGBoost classifier attained a sensitivity of 95.48% and a specificity of 84.93%, indicating strong effectiveness in correctly identifying patients with liver disease while preserving balanced detection of healthy cases. Random Forest achieved a slightly higher sensitivity of 97.18%, but this improvement came at the cost of reduced specificity (80.14%),

reflecting a greater tendency to misclassify healthy individuals. In contrast, Gradient Boosting and Decision Tree exhibited comparatively weaker performance on both metrics. Models such as Support Vector Machine and Logistic Regression, which demonstrated particularly low specificity, were deemed less suitable for clinical application, as reliable discrimination between diseased and non-diseased cases is essential for medical decision-making.

Table 2: Model Comparison with Sensitivity and Specificity

Model	Test Accuracy	Sensitivity	Specificity
XGBoost	0.924	0.9548	0.8493
Random Forest	0.922	0.9718	0.8014
Gradient Boosting	0.866	0.9435	0.6781
Decision Tree	0.854	0.8955	0.7534
K-Nearest Neighbors	0.832	0.8870	0.6986
Logistic Regression	0.712	0.9294	0.1849
Support Vector Machine	0.708	1.0000	0.0000
Naïve Bayes	0.556	0.4124	0.9041

Further optimization of the XGBoost model through hyperparameter tuning resulted in an improved classification accuracy of 93.4%, as reported in the classification summary. For the liver disease class, the model achieved an F1-score of 0.95, reflecting a strong balance between precision and recall. Such equilibrium is particularly critical in clinical contexts, where minimizing false negatives—patients incorrectly classified as healthy—is essential to ensure timely diagnosis and safeguard patient outcomes.

Classification Report (Tuned XGBoost Model)

- Accuracy: **0.934**
- Precision (Liver Disease = 1): **0.94**
- Recall (Liver Disease = 1): **0.97**
- F1-score (Liver Disease = 1): **0.95**
- Precision (Non-Liver = 0): **0.91**
- Recall (Non-Liver = 0): **0.86**
- F1-score (Non-Liver = 0): **0.88**

The incorporation of SHapley Additive Explanations (SHAP) provided model interpretability by quantifying the relative contribution of each feature to predictive outcomes. The global SHAP analysis revealed that Alkaline Phosphatase, Direct Bilirubin, Total Bilirubin, and the Albumin-to-Globulin Ratio were the most influential variables in distinguishing patients with liver disease. In addition, local SHAP explanations delivered patient-specific insights, allowing clinicians to understand the rationale behind individual predictions and thereby facilitating more informed medical decision-making.

In contrast to earlier studies that focused predominantly on maximizing predictive accuracy [1–5], the proposed system demonstrates both high classification performance and interpretability. This combination strengthens clinical reliability, enhances practitioner confidence, and establishes the system as a viable decision-support tool for liver disease diagnosis.

6. CONCLUSIONS

This endeavour effectively created a robust and user-friendly system for the early identification of liver disease using a machine learning-based approach. This project uses the XGBoost method to create a robust model capable of forecasting liver illness with high precision using patient details and test values. The model was trained on a cleaned dataset, where values that were missing were filled in and the data was adjusted to improve performance. The model was then connected to a Django web application. The web app has a simple front page where users can enter medical details, get prediction results, check past records, and manage their data easily. SHAP is also included to make the results clearer. It shows how each patient detail affects the prediction, which helps doctors and users understand the outcome and trust the system more. Overall, the system meets its goals of being accurate, easy to use, secure, and scalable. It gives fast and accessible liver disease screening,

making it useful even in remote areas or places has little access to healthcare resources, while also supporting better decision-making for healthcare professionals.

REFERENCES

[1] P. Rajeswari and S. Reena, “Analysis of Liver Disorder Data Set Using Classification Algorithms,” International Journal of Computer Applications, vol. 1, no. 4, pp. 1–4, 2010.

[2] J. Singh and K. Kaur, “Prediction of Liver Disease Using Classification Techniques,” International Journal of Computer Science and Information Technologies, vol. 7, no. 3, pp. 1174–1177, 2016.

[3] S. Kalra, A. Singla, and N. Arora, “Liver Disease Prediction Using Machine Learning Techniques,” International Journal of Engineering and Advanced Technology, vol. 8, no. 6, pp. 224–227, 2018.

[4] S. Choubey and A. Paul, “An Efficient Model for Liver Disease Diagnosis Using Ensemble Learning,” International Journal of Innovative Technology and Exploring Engineering, vol. 9, no. 3, pp. 1502–1506, 2020.

[5] R. Vaidya and M. S. Patil, “Liver Disease Detection Using XGBoost Classifier,” International Journal of Research in Engineering, Science and Management, vol. 5, no. 6, pp. 94–97, 2022.

[6] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining (KDD), pp. 785–794, 2016.

[7] S. Lundberg and S. Lee, “A unified approach to interpreting model predictions,” Proc. 31st Conf. Neural Information Processing Systems (NeurIPS), pp. 4765–4774, 2017.