Cureus Journal of Computer Science

Open Access Original Article

A Fast and Reliable Transformer-Based TabPFN Model for Liver Disease Diagnosis

Pradeep Rawat [1] , Prabh Deep Singh [1] , Vikas Tripathi [1]

1. Department of Computer Science and Engineering, Graphic Era (Deemed to be University), Dehradun, IND

Corresponding authors: Pradeep Rawat, pradeeprawat.cse@gmail.com, Prabh Deep Singh, ssingh.prabhdeep@gmail.com, Vikas Tripathi, vikastripathi.cse@geu.ac.in

Received 03/25/2025
Review began 04/07/2025
Review ended 05/13/2025
Published 05/15/2025

DOI: https://doi.org/10.7759/s44389-025-04072-8

Abstract

Early and accurate diagnosis of liver disease is the key to timely treatment and improved patient outcomes. Traditional machine learning models often utilize manually constructed features and struggle to extrapolate to high-dimensional, complex tabular data. This work presents an original approach to predicting liver disease using TabPFN (Prior-Data Fitted Network for Tabular Data), a pre-trained transformer-based model on artificial tabular problems. This model incorporates one-dimensional feature attention and one-dimensional sample attention techniques into the model, and subsequently a multilayer perceptron classifier. Experimental results on the Indian Liver Patient Dataset demonstrate better performance than existing approaches, with an accuracy of 90.36%, an F1-score of 90.35%, a precision of 90.99%, and a recall of 90.36%. These results show how the suggested architecture model effectively captures relevant patterns in tabular information for liver disease identification.

Categories: AI applications, Health Informatics, Machine Learning (ML)
Keywords: tabpfn, liver disease prediction, machine learning, medical diagnostics, tabular data classification, small dataset classification, transformers in healthcare, meta-learning

Introduction

Liver disease is a significant public health issue worldwide, causing over 2 million deaths annually. It is responsible for one death in every 25, or 4% of all mortality worldwide. There are many diseases, including cirrhosis, viral hepatitis, liver cancer, and non-alcoholic fatty liver disease (NAFLD), that cause the burden of liver disease. Environmental, genetic, and lifestyle-related factors all play a part in global liver disease prevalence. Excessive drinking of alcohol is a major risk factor for liver disease, disability, and premature death. Because 43% of the world's population consumes alcohol, alcohol-related liver disease has an important role to play. About 6.4 L of alcohol was consumed per year per person among people who were 15 years and older in 2016. Alcohol misuse has damaging consequences that extend beyond liver disease; it also leads to a variety of other diseases that increase the overall burden of the disease. Moreover, alcohol use disorder impacts nearly 283 million individuals worldwide, with a prevalence of 5.1%. Moreover, NAFLD, a disease strongly linked with metabolic diseases such as obesity and type 2 diabetes, impacts nearly 25% of adults globally. Around 1.1 million individuals lost their lives due to viral hepatitis B- and C-related conditions in 2020 alone, highlighting the imperative for early detection and effective management methods [1].

Liver illness development and progression are determined by several factors. Obesity, viral infections, hereditary pre-dispositions, type 2 diabetes, and unsafe medical procedures, including needle sharing, are recognized as major causative factors. Approximately 1.5 billion cases of chronic liver disease were reported in 2017 alone throughout the world [2], constituting nearly 19.7% of the population. The increasing incidence of liver disease highlights the need for reliable and readily accessible diagnostic tools as a matter of real urgency. The most valid means of detecting liver disease remain the conventional diagnostic methods like bloodwork, liver biopsy, computed tomography scans, ultrasounds, and magnetic resonance imaging. Yet, these methods are often time-consuming, require specialized training for accurate interpretation, and may not be readily accessible within rural or resource-poor settings. Consequently, numerous patients have a late diagnosis, which exacerbates clinical outcomes and leads to the progression of the disease. The capability of analyzing and comprehending biological information has been enhanced significantly through the integration of bioinformatics and computer tools, and this has made the identification and diagnosis of liver diseases better [3]. Improved patient outcomes can be achieved through the application of computational techniques in medical diagnosis to enhance the accuracy and efficiency of disease identification.

Diagnosis and management of liver disease have been greatly improved by advances in artificial intelligence (AI). The capacity of machine learning (ML) algorithms to sift through medical records, identify trends, and detect patterns indicative of liver disease has been remarkable [4]. AI-powered methods enable automated analysis, reducing the requirement for human interpretation and enhancing the accuracy of diagnosis. Despite all these advances, there are some limitations in applying ML models to

medical diagnoses. To begin with, datasets such as the Indian Liver Patient Dataset (ILPD) [5] are limited in size because they contain very few 583 patient records and 10 clinical indicators. The extremely low sample size presents a challenge to developing stable models that can successfully generalize to unseen data. In addition, clinical, biochemical, and demographic data are all found in medical databases, which by their very nature are diverse and do not contain any spatial or temporal association. Due to such complexities, specialized ML algorithms that suit the specificity of medical databases have to be established. Second, the lack of medical data often leads to conditions such as overfitting in standard ML algorithms, including ensemble methods, logistic regression (LR), and support vector machine (SVM). These models are less applicable for actual use cases with limited data access because they require significant hyperparameter fine-tuning and tuning to achieve optimal performance.

New approaches to the processing of tabular data have been enabled by recent advances in deep learning. For instance, TabNet [6] has proven useful in enhancing the analysis of tabular datasets. Nevertheless, to achieve peak performance, deep learning models often require large datasets with more than 10,000 samples and considerable processing power. Small medical datasets are not well-suited for deep learning models based to their heavy computational cost and data demand. Hollmann et al. [7] proposed TabPFN (Prior-Data Fitted Network for Tabular Data), based on transformer architecture and meta-learning approaches, to address these limitations. TabPFN is pre-trained on over 100 million artificial datasets [8], which enables it to generalize effectively to novel datasets without needing a lot of hyperparameter adjustment compared to conventional ML models. TabPFN allows for fast and accurate inferences in one forward pass by combining past information through learned feature transformations. Due to this ability, it is particularly suited for medical use, where timely and accurate predictions are critical for treatment planning and patient monitoring.

This research is conducted using the ILPD dataset [5], which contains 583 patient records and 10 clinical indicators, such as bilirubin and liver enzyme levels. The primary objectives of this study are to assess TabPFN's performance in liver disease classification based on the ILPD dataset and compare it with existing models. From this study, one can see that TabPFN has the potential to improve liver disease prediction, particularly for use in clinics where there are few resources. AI-based strategies such as TabPFN can contribute towards better patient outcomes and the minimization of healthcare system burden in the wake of liver disease worldwide through enhanced detection and intervention earlier on.

Materials And Methods

Literature survey

There has been a lot of interest in predicting liver disease in recent times with ML and deep learning (DL) techniques. Various algorithms have been investigated in various studies, each with its merits and demerits, to ameliorate diagnostic correctness and dependability.

Srivastava et al. [9] evaluated different ML strategies, namely naive Bayes, K-nearest neighbors (KNN), and LR. Among all the models evaluated, the study found that LR achieved the highest accuracy at 75%.

Gamara et al. [10] suggested an ML-based classification that uses the output of medical tests to classify liver disease. Following their study, the SVM model achieved 79.5% accuracy, while artificial neural network (ANN) achieved the highest accuracy of 89%. The study suggested future research with larger and better-balanced datasets and stressed that AI-based models can complement medical professionals rather than replace them.

Yadav et al. [11] compared the performance of DL models like convolutional neural network (CNN), long short-term memory (LSTM), and CNN-bidirectional LSTM (BiLSTM) models with respect to classification accuracy. They reported that the CNN-BiLSTM model outperformed individual CNN and LSTM models, with a test accuracy of 84%.

Mohamed et al. [12] compared several ML models like random forest (RF), gradient boosting classifier, SVM, LR, and decision tree classifier using the Kaggle Indian Liver Patient Dataset. In the study, RF achieved the highest accuracy among the evaluated methods when combined with PCA for feature selection, reaching an accuracy of 80.34%.

Rao et al. [13] compared ANN with other models like KNN, RF, LR, and SVM. According to their results, after performing feature selection along with hyperparameter tuning, both SVM and ANN performed well; the F1-score achieved 83.38% with 73.57% accuracy, whereas LR achieves the greatest value of the F1-score of 83.85% without doing feature selection.

Minnoor et al. [14] discussed the efficiency of a few supervised learning techniques, including multilayer perceptron (MLP), LightGBM, extra trees, KNN, and LR. As found in their experiment results, the extra trees classifier could perform best in the classification of liver disease, with a maximum accuracy of up to 88.94% and an F1-score of 0.88.

Anthonysamy et al. [15] evaluated classification models for the prediction of liver disease with Deep Neural Network Multilayer Perceptron, SVM, KNN, and Hard Voting Classifier. The findings of the study revealed that alongside 0.78 accuracy, 0.80 precision, and 0.87 F1-score, the model of Hard Voting Classifier outscored all others. Moreover, the MLP was ranked second, along an F1-score of 0.87 and an accuracy of 0.77.

Kumar et al. [16] demonstrated how Bayesian optimization improved classifier performance in the diagnosis of liver disease. The experiment compared the variation between RF, SVM, Adaptive Boosting (AdaBoost), and Extreme Gradient Boosting (XG Boost). Bayesian optimization was applied to optimize key parameters, whereas Pearson correlation was utilized to select features. Training and testing were conducted on a Kaggle dataset from the UCI Machine Learning Repository. The research concludes that optimization enhanced the accuracy of classification, and RF performed better than SVM (80.81%), XGBoost (79.85%), and AdaBoost (77.08%), with the best performance recorded as 81.06%.

Rani et al. [17] suggested predicting liver disease based on a semi-supervised method that integrates SVM and centroid-based clustering. SVM initially predicts whether the patient suffers from liver disease and, in case of a positive result, centroid-based clustering predicts the severity of the disease. The UCI dataset was used by the study to present diagnosis and severity ratings along with treatment suggestions in a bid to lighten the load of physicians. Compared with other models, the results indicated that hybrid SVM with centroid-based clustering was better (89% accuracy), followed by Hybrid SVM + K-Means (80%) and K-Means (76%).

Riya et al. [18] employed a database of 416 Indian hospital patients to design risk prediction models for liver disease. Decision trees, RF, and LR were compared for their accuracy, specificity, sensitivity, and area under the receiver operating characteristic (ROC) curve. The study concluded that, with an area under the curve (AUC) of 0.80, 77.2% specificity, 71.4% sensitivity, and 75.7% accuracy, the hybrid model was superior among the three.

Zhao et al. [19] have presented the W-LR-XGB algorithm to carry out initial screening work as well as to predict liver disease. Using 573 liver function data samples from patients, the authors compared the performance of W-LR-XGB with baseline models and found CatBoost to be a better approach. With the model having 81% F1-score, 82% recall, and 83% accuracy in making predictions, it showed good performance.

Amin et al. [20] suggested a feature extraction method for the classification of liver disease patients using an ensemble of ML classifiers, including LR, RF, KNN, SVM, MLP, and a voting ensemble classifier. To obtain the maximum classification accuracy, feature extraction was performed after pre-processing the data to handle outliers and missing values. The system attained an F1-score of 88.68%, an AUC of 88.20%, 88.10% accuracy, 85.33% precision, and 92.30% recall.

The collective findings of these studies, also shown in Table *1*, indicate that ML and DL models both play important roles in predicting liver disease, with ANN, Extra Trees, and Hard Voting Classifiers performing best on different datasets and situations. Model accuracy is significantly improved through the use of feature selection methods like PCA, while improving performance is highly dependent on hyperparameter tuning. Different studies have indicated that total accuracy of liver disease prediction is between 64.4% and 89%, depending on the utilized ML or DL model. The highest accuracy levels were obtained by ANN (89%) [10], Extra Trees (88.94%) [14], and CNN-BiLSTM (84%) [11]. Alarmingly, though, the leading limitation of these studies is the small sample size of ILPD, consisting of only 583 patient records. The relatively small dataset may result in skewed predictions and constrain the general applicability of the results. Succeeding studies should address larger, representative datasets to improve model robustness and diagnostic reliability.

Study	Models Evaluated	Best Model	Accuracy (%)	Other Metrics
[9]	Naïve Bayes, KNN, Logistic Regression	Logistic Regression	75	–
[10]	SVM, ANN	ANN	89	–
[11]	CNN, LSTM, CNN-BiLSTM	CNN-BiLSTM	84	–
[12]	Random Forest, Gradient Boosting, SVM, Logistic Regression, Decision Tree	Random Forest (PCA)	80.34	–
[13]	ANN, KNN, Random Forest, Logistic Regression, SVM	SVM	73.57	F1-score: 83.85 (LR), 83.38 (SVM)
[14]	MLP, LightGBM, Extra Trees, KNN, Logistic Regression	Extra Trees	88.94	F1-score: 0.88
[15]	MLP, SVM, KNN, Hard Voting Classifier	Hard Voting Classifier	78	Precision: 0.80, F1-score: 0.87
[16]	Random Forest, SVM, AdaBoost, XGBoost	Random Forest	81.06	SVM: 80.81, XGBoost: 79.85, AdaBoost: 77.08
[17]	Hybrid SVM, K-Means, Centroid-Based Clustering	Hybrid SVM + Centroid-Based Clustering	89	Hybrid SVM + K-Means: 80, K-Means: 76
[18]	Decision Tree, Random Forest, Logistic Regression	Hybrid Model	75.7	AUC: 0.80, Sensitivity: 71.4, Specificity: 77.2
[19]	W-LR-XGB, CatBoost	CatBoost	83	Recall: 82, F1-score: 81
[20]	Logistic Regression, Random Forest, KNN, SVM, MLP, Ensemble Voting	Ensemble Voting	88.10	Precision: 85.33, Recall: 92.30, F1-score: 88.68, AUC: 88.20

TABLE 1: Comparison of different models for disease prediction.

KNN, K-Nearest Neighbors; SVM, Support Vector Machine; ANN, Artificial Neural Network; CNN, Convolutional Neural Network; LSTM, Long Short-Term Memory; BiLSTM, Bidirectional Long Short-Term Memory; MLP, Multilayer Perceptron; W-LR-XGB, Weighted Logistic Regression combined with Extreme Gradient Boosting; PCA, Principal Component Analysis; LR, Logistic Regression; AUC, Area Under the Curve

Proposed methodology

Principal Objective

The primary goal of this research is to develop a precise and efficient predictive model for liver disease using the TabPFN classifier. Liver disease is a significant global health concern, and early diagnosis plays a crucial role in improving patient outcomes. Unlike conventional ML models that require extensive hyperparameter tuning, TabPFN offers fast inference and high accuracy with minimal computational effort, making it an ideal choice for this task. This study follows a structured ML methodology, incorporating exploratory data analysis, preprocessing, and the application of advanced ML techniques to effectively classify individuals into healthy and diseased categories based on a set of clinical parameters.

Data Collection

The ILPD of the UCI Machine Learning Repository [5], containing 583 patient records and clinical data relevant to the diagnosis of liver disease, was used as the dataset for this study. It contains biochemical markers that give insight into liver function, including albumin, protein levels, total bilirubin, aspartate aminotransferase (AST), and alanine aminotransferase (ALT). The target variable represents the diagnosis 1 for healthy and 2 for diseased. Every sample in the set is expressed as a feature vector:

For dataset D:

$$D = \{(x_i, y_i) \mid x_i \in \mathbb{R}^m, y_i \in \{1, 2\}, i = 1, 2, ..., n\}$$

where x_i is the feature vector of the i-th patient and y_i is the respective diagnosis label (1 for healthy, 2 for diseased).

Exploratory Data Analysis

Knowledge of the dataset structure, missing value detection, feature distribution assessment, and identification of biases all rely on exploratory data analysis. We can make informed decisions regarding preprocessing and feature selection by demonstrating feature and target variable relationships.

1) Identifying Missing Values: Many reasons, including measurement error, non-responsiveness of patients, and inadequate records, may lead to missing values in clinical datasets. Before the selection of an appropriate handling strategy, it is crucial to detect and understand the distribution of missing values (Figure *1*). The detection of patterns, e.g., whether missingness is random or systematically related to specific traits, becomes simpler using a visual representation of missing data. For each feature, we define a binary indicator function to quantify missingness:

$$M_{ij} = \begin{cases} 1, & \text{if } x_{ij} = \text{NaN} \\ 0, & \text{otherwise} \end{cases}$$

where M_{ij} indicates the presence (1) or absence (0) of missing data in feature j for sample i. The total missing values for each feature can be calculated using

$$M_j = \sum_{i=1}^{n} M_{ij}$$

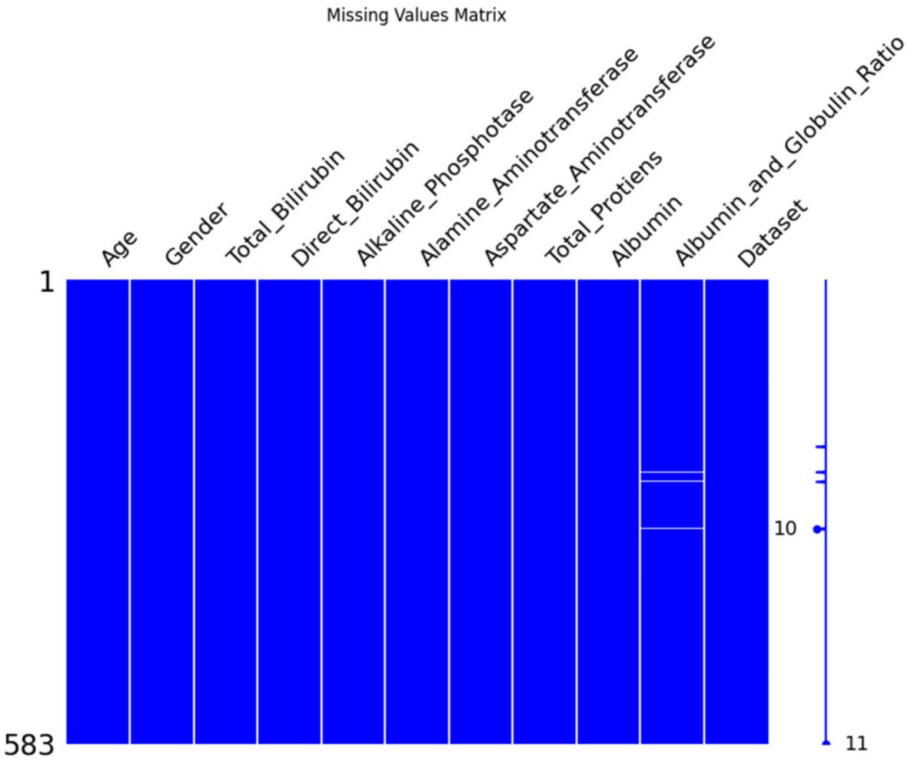

FIGURE 1: Heatmap showing the distribution of missing values across features. Lighter areas indicate missing values, revealing patterns in missingness.

We can determine whether to drop or impute missing values in subsequent preprocessing steps by looking at the missingness structure. This dataset has four missing values.

2) Feature-Target Relationship: Understanding predictive patterns requires the analysis of the relationship between clinical measurements and the presence of disease. The values of liver enzymes, such as bilirubin, albumin, ALT, and AST, often have varying distributions among healthy and ill patients. Significant insights into these interactions can be obtained by a careful correlation analysis (Figure *2*).

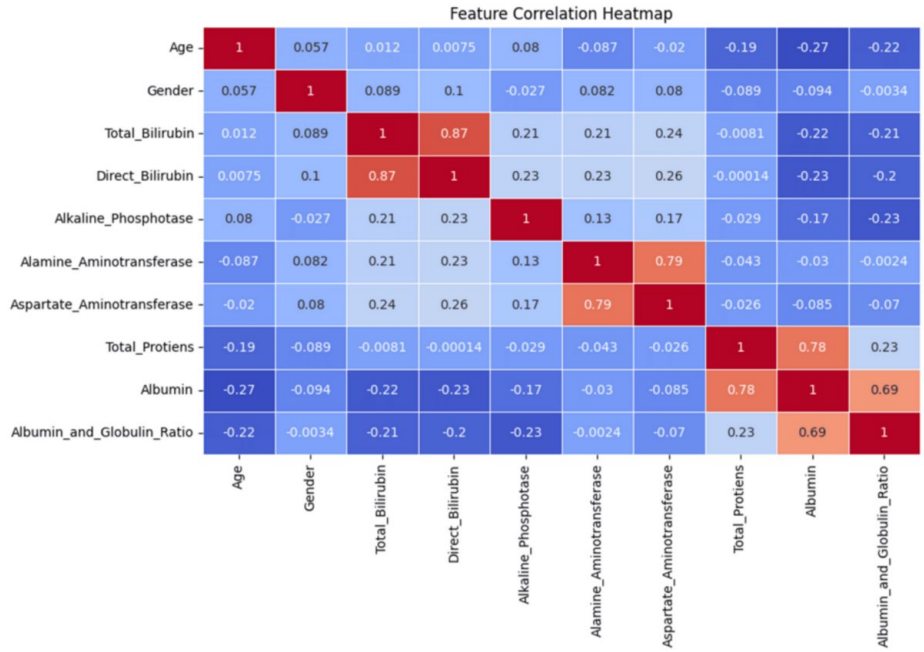

FIGURE 2: Feature correlation heatmap illustrating relationships among clinical parameters and their correlation with liver disease.

Key insights from correlation analysis are discussed as follows:

• Strong Positive Correlations (Multicollinearity Issues)

Total Bilirubin and Direct Bilirubin (0.87): Having both might introduce redundancy since direct bilirubin is a subset of total bilirubin.

ALT and AST (0.79): These two liver enzymes are strongly correlated, so one could be removed to reduce multicollinearity.

Albumin and Total Proteins (0.78): There is a very strong relationship between the two since albumin contributes a large percentage of total proteins.

Albumin and Albumin/Globulin Ratio (0.69): Albumin concentration directly influences the albumin/globulin ratio.

• Weak or No Correlation With Age

All the variables, such as dataset (Liver Disease Diagnosis), had weak correlations with age (-0.14), meaning that liver disease cannot be highly predicted based on age.

• Relationship Between Features and Liver Disease (Dataset)

Albumin (0.16) and albumin/globulin ratio (0.16) values are most positively correlated with liver disease, which means that low values are possibly associated with the disease.

Direct bilirubin (-0.25) and total bilirubin (-0.22) have negative correlations, which reflect the fact that liver disease patients often have high levels of bilirubin.

ALT (-0.16) and AST (-0.15) have weakly negative correlations, as one would expect of markers of liver damage.

• No Strong Correlation With the Target Variable

The correlations within the dataset are relatively weak (between 0.15 and - 0.25), which means that no single trait is a good predictor by itself.

To make an accurate prediction, a machine-learning algorithm must depend on a combination of features.

3) Class Imbalance: Class imbalance is one of the significant issues in medical data sets, wherein there are way more healthy patients than patients suffering from any condition. The performance of the model can be biased by this class imbalance. Disproportionate representation is shown by visualizing class distributions with the help of bar charts (Figure 3).

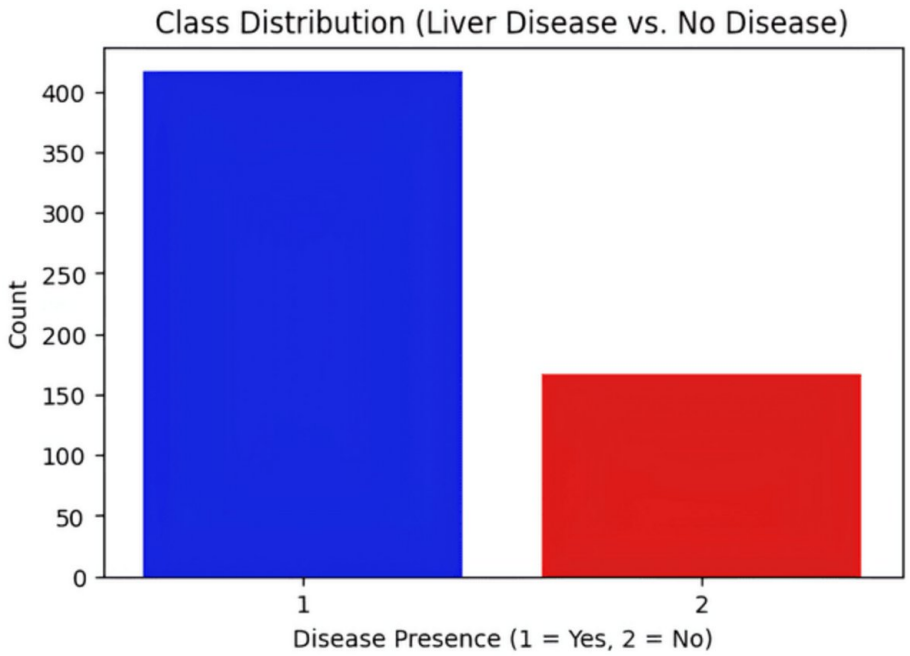

FIGURE 3: Class distribution of the dataset showing an imbalance between healthy and diseased samples.

Preprocessing

The cleanliness, organization, and trainability of the dataset are ensured through preprocessing. Missing data handling, feature selection of relevant features, categorical variable encoding, class balance, and feature scaling are the key steps.

1) Handling Missing Values: Missing values in medical datasets can potentially cause bias and reduce model reliability. In this research, missing entries are removed instead of imputing missing values. Two reasons justify this approach:

• Preserving Data Integrity: Since clinical measurements often rely on each other, statistical methods (e.g., mean or median imputation) for missing values may introduce spurious associations that do not exist. Interpretability can be affected, and the model can be deceived.

• Minimizing Noise and Uncertainty: Clinical data are highly sensitive, and missing values might indicate root system issues (e.g., unrecorded tests due to patient-specific issues). Incorrect conclusions about the health status of the patient might arise from the imputation of missing variables.

Thus, list-wise deletion was applied, which removes any sample with missing values:

$$D' = \{(x_i, y_i) \mid \forall j, x_{ij} \neq \text{NaN}\}$$

where D' is the cleaned data set that contains only complete observations.

The model learns from high-quality, well-defined data by ensuring that only complete records are utilized, which reduces the chance of incorrect predictions.

2) Feature Selection and Target Definition: During feature selection, the independent variable and dependent variable are distinguished to demarcate the feature matrix X and the target vector y:

$$X = D' \setminus \{\text{Dataset}\}, \quad y = D'[\text{Dataset}]$$

where 'Dataset' denotes the target variable, which is a class label. Aside from the deletion of 'Dataset', no other features are dropped.

3) Encoding Categorical Variables: Gender is a categorical attribute in the dataset and has to be numerically represented. Utilizing label encoding:

$$\phi(\text{Male}) = 1, \quad \phi(\text{Female}) = 0$$

This transformation ensures compatibility with ML models.

4) Handling Class Imbalance: Upsampling is employed to correct class imbalance and ensure a fair representation of both classes. The minority class is duplicated until its size is the same as that of the majority class (Figure 4):

$$C_2'| = |C_1|$$

where C_1 is the number of samples in the majority class, C_2 is the number of samples in the minority class, and C'_2 is the number of samples in the resampled minority class, which is generated by resampling C_2:

$$C_2' = \{x_j^* \sim C_2 \mid j = 1, ..., |C_1|\}$$

where x_j^* is a resampled instance drawn from the minority class C_2.

This method improves classification performance and prevents the model from being biased toward the majority class.

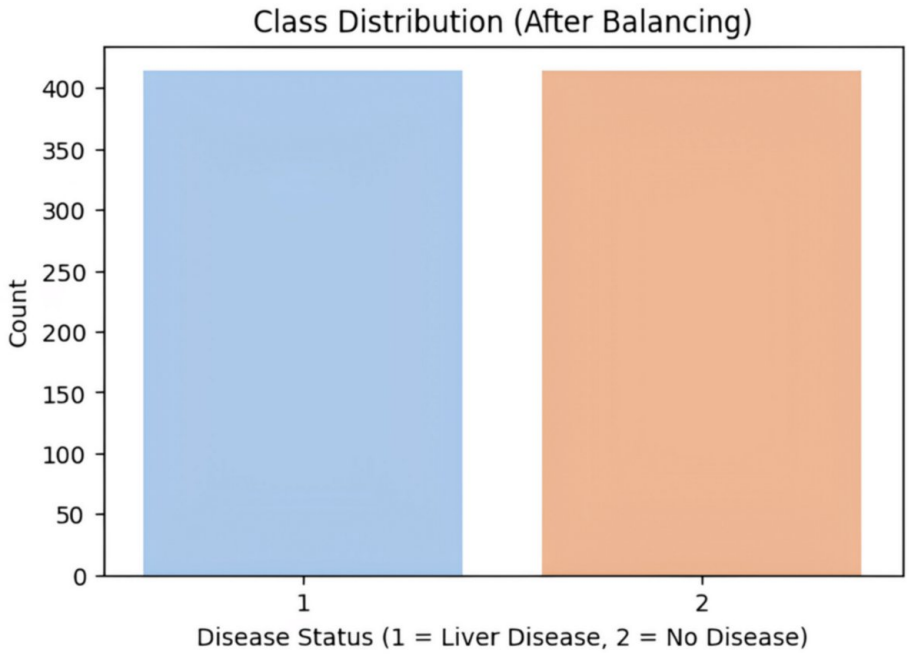

FIGURE 4: Class distribution after balancing.

5) Train-Test Split: The data is split into training and testing subsets for evaluating model performance:

$$X_{\text{train}}, X_{\text{test}}, y_{\text{train}}, y_{\text{test}} = \text{split}(D'', \text{test size} = 0.2)$$

where D'' is the processed dataset, X_{train} and X_{test} are the training and testing features, y_{train} and y_{test} are the corresponding labels, and a stratified split with mtest size 0.2 is used to preserve class distribution.

6) Feature Distribution and Standardization: Understanding data properties and model behavior requires

feature distribution analysis. Outliers exist in most clinical markers, especially liver enzymes, which have considerable fluctuations. Leveling feature magnitudes through standardization ensures numerical stability and improves optimization performance.

a) Feature Distribution and Outliers: Before preprocessing, features show significant variability. As illustrated in Figure 5, liver enzyme levels such as alkaline phosphatase, ALT, and AST display extreme outliers, suggesting the presence of patients with severe liver conditions. Most other features have lower variance, indicating more stable distributions.

Features are highly variable before preprocessing. There are extreme outliers in the levels of alkaline phosphatase, ALT, and AST, as demonstrated in Figure 5, representing the existence of patients with critical liver issues. The variability of most other features is less, implying more consistent distributions.

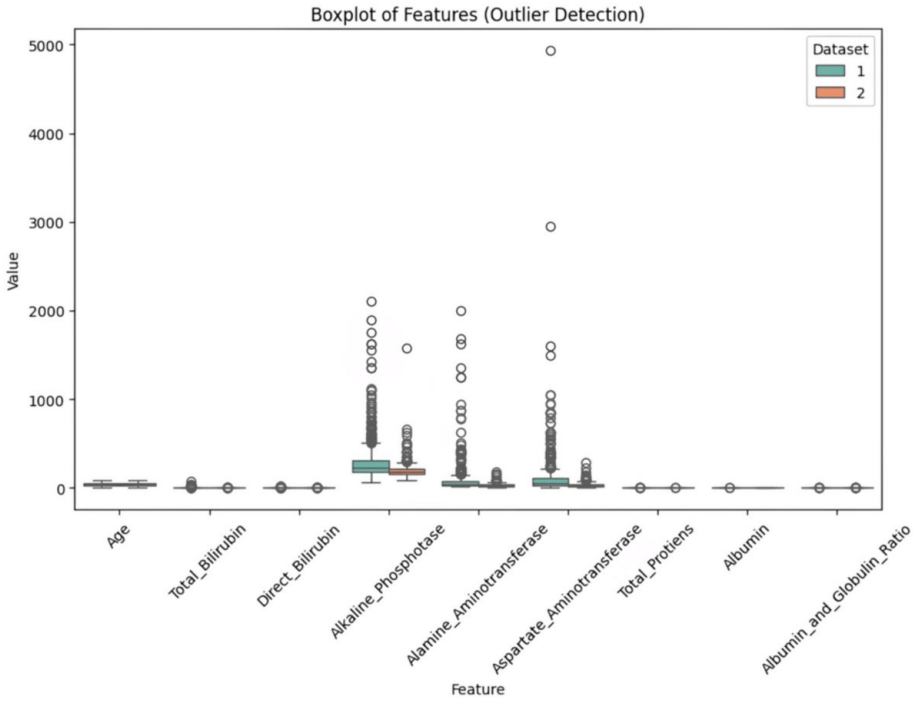

FIGURE 5: Boxplot of raw feature distributions, highlighting outliers in liver enzyme levels.

b) Feature Standardization: Standardization is applied because ML models can be sensitive to feature scales:

$$x_j' = \frac{x_j - \mu_j}{\sigma_j}$$

where μ_j and σ_j represent the mean and standard deviation of feature x_j in the training set, and x_j' is the standardized feature with zero mean and unit variance.

By ensuring each feature has a zero mean and unit standard deviation, this transformation prevents features with large scales from overpowering.

After standardization, all features are normalized into a comparable scale, as depicted in Figure 6, enhancing their comparability. Outliers are still there, but they have less of an effect on model training.

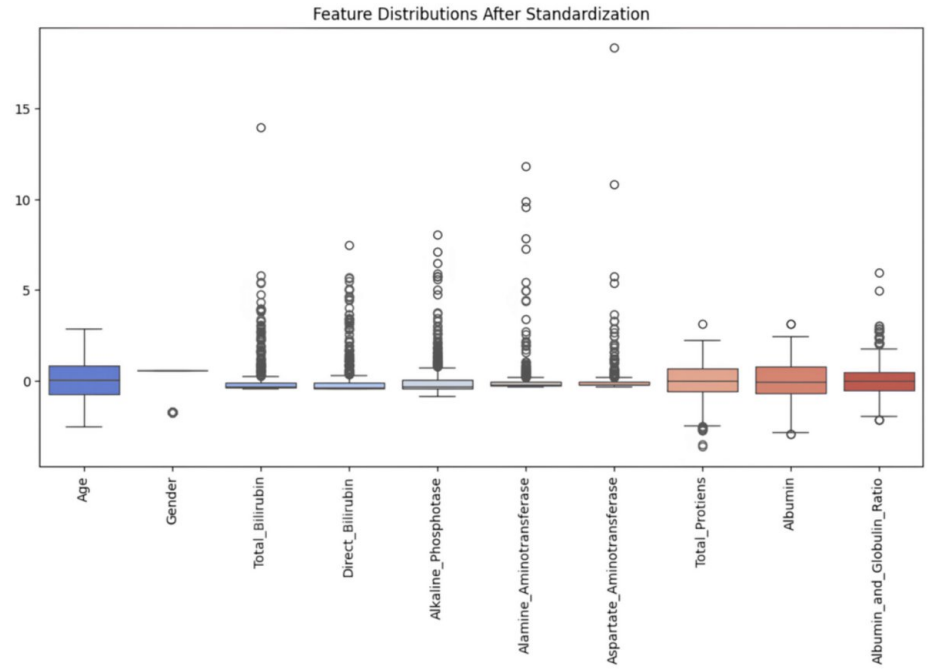

FIGURE 6: Feature distributions after standardization, ensuring uniform scaling while preserving outliers.

TabPFN Classifier-Detailed Working

The updated dataset is input to the TabPFN classifier, an innovative transformer-based tabular data classifier, after preprocessing. Unlike traditional ML models that require one-pass inference or iterative training and hyperparameter adjustment, TabPFN employs a transformer framework that is already pre-trained and intended for one-pass inference. By acquiring a previous distribution over possible functions and directly predicting class probabilities for future samples in one forward pass, the method casts classification as a Bayesian inference problem. The architecture model of TabPFN is shown in Figure 7.

1) Internal Architecture of TabPFN: The following key components constitute the transformer-based layer stack that TabPFN internally employs to process input data:

Feature Projection Mechanism: TabPFN works with raw tabular features within the transformer model, in contrast to typical transformers that rely on an embedded layer. Self-attention mechanisms implement linear transforms for internally projecting and normalizing input features into higher dimensions as opposed to static embeddings. This ensures dynamic learning of feature interaction without the use of explicit feature embeddings.

1D Feature Attention Layer: For enhanced discriminative feature learning, this layer adjusts importance weights dynamically by computing correlations among input features. It focuses on feature relevance over sequence order, unlike traditional NLP-based attention algorithms. The formula for computing the attention weights is

$$\alpha_i = \mathrm{Softmax}\left(\frac{Q_i K_i^T}{\sqrt{d}}\right) V_i$$

where d is the dimension of the feature Q, K, and V, which are query, key, and value matrices, and α_i represents the attention-weighted output for the i_{th} input.

1D Sample Attention Layer: Generalizing over different data distributions, this layer improves the classification probabilities by evaluating relationships over different input samples. Rather than focusing solely on the importance of single features, it helps the model to determine trends over different occurrences.

Multi-Layer Perceptron (MLP) Classifier: A connected MLP further enhances feature interactions across several layers with activation functions after being given the output of the attention layers. The

transformation is as follows:

$$h = \sigma(Wx + b)$$

where *x* is the input vector, *W* and *b* are the learnable weight matrix and bias, σ is a non-linear activation function (e.g., ReLU or GELU), and *h* is the transformed feature representation.

Probabilistic Output Layer (Bayesian Inference and Class Prediction): TabPFN produces a probability distribution over class labels, unlike traditional classifiers, which produce deterministic labels. The model employs the Bayesian probability estimate:

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)}$$

where:

• P(Y|X) represents class *Y* posterior probability, given feature vector *X*.
• P(X|Y) is the likelihood function, modeling how likely *X* is for a given class *Y*.
• P(Y) is the prior probability of class distribution.
• P(X) is a normalizing constant ensuring a valid probability distribution.

The maximum A posteriori (MAP) estimate is applied to compute the final class prediction:

$$\hat{Y} = \arg\max_{Y} P(Y|X)$$

where $\hat{Y}$ denotes the predicted class label obtained by selecting the class *Y* that maximizes the posterior probability *P*(*Y* | *X*).

By ensuring the highest posterior probability class is selected, this choice rule makes the process of classification less sensitive to uncertainty and more naturally probabilistic.

TabPFN works exceptionally well for such medical classification issues as liver disease diagnosis due to this technique's significant elimination of overfitting and generalization improvement.

Rationale for Using TabPFN

Since TabPFN can be well generalized on tabular datasets without any hyperparameter tuning, it was selected. TabPFN is compared with traditional models in the following Table 2.

Feature	Traditional ML Models	TabPFN
Hyperparameter Tuning	Required	Not Required
Training Time	High	Low
Performance on Small Datasets	Moderate	High
Interpretability	Low to Moderate	High
Scalability	Limited	Efficient

TABLE 2: TabPFN compared with traditional ML models.

ML, Machine Learning; TabPFN, Prior-Data Fitted Network for Tabular Data

Algorithm for TabPFN Model for Liver Disease Classification

Require: Dataset D = (X, y) with categorical and numerical features

Ensure: Predicted labels $\hat{y}$ for test data

1: Step 1: Handle Missing Values

2: Remove instances with missing values:
$X = X_{\text{cleaned}} = \{x_i \mid x_i \neq \text{NaN} \, \forall i\}$

where x_i represents each data instance and NaN indicates missing values.

3: Step 2: Encode Categorical Features
4: Apply label encoding to the categorical feature Gender:

$$X_{\text{Gender}} = \begin{cases} 1, & \text{if Male} \\ 0, & \text{if Female} \end{cases} \#$$

where X_{Gender} is the numerical encoding of the Gender feature.

5: Step 3: Handle Class Imbalance
6: Separate majority and minority classes:
$D_1 = \{(X_i, y_i) \mid y_i = 1\}, \quad D_2 = \{(X_i, y_i) \mid y_i = 2\}$

where D_1 and D_2 represent subsets of the data with class labels 1 and 2, respectively.

7: Apply upsampling to the minority class:
$D_2' \sim D_2 \quad \text{such that } |D_2'| = |D_1| \#$

where D_2' is the resampled version of the minority class D_2, sampled with replacement to match the size of D_1.

8: Combine to obtain a balanced dataset:
$D' = D_1 \cup D_2'$

where D' is the balanced dataset created by combining the majority class with the upsampled minority class.

9: Step 4: Train-Test Split
10: Split dataset into training and test sets:
$(X_{\text{train}}, X_{\text{test}}, y_{\text{train}}, y_{\text{test}}) = \text{train\_test\_split}(X, y) \#$

where X_{train}, X_{test} are training and testing features, and train, y_{test} are the corresponding labels.

11: Step 5: Feature Scaling
12: Apply standardization:

$$X' = \frac{X - \mu}{\sigma}$$

where μ and σ are the mean and standard deviation of each feature in the training set, and X' is the standardized feature matrix.

13: Step 6: TabPFN Classification
14: Initialize TabPFN classifier f_θ.

15: Train the model:
$f_\theta \leftarrow \text{train}(X_{\text{train}}, y_{\text{train}})$

where f_θ represents the TabPFN model with parameters θ, trained on the training data.

16: Predict on test data:
$\hat{y} = f_\theta(X_{\text{test}})$

where $\hat{y}$ is the predicted label vector for the test set.

17: Step 7: Model Evaluation
18: Compute accuracy:

$$\text{Accuracy} = \frac{\sum_{i=1}^{N} \mathbb{1}(\hat{y}_i = y_i)}{N}$$

where N is the total number of test samples, $\hat{y}_i$ is the predicted label, y_i is the true label, and $\mathbb{1}$ is the indicator function returning 1 if the prediction is correct.

19: Generate classification report.

Model Architecture Diagram

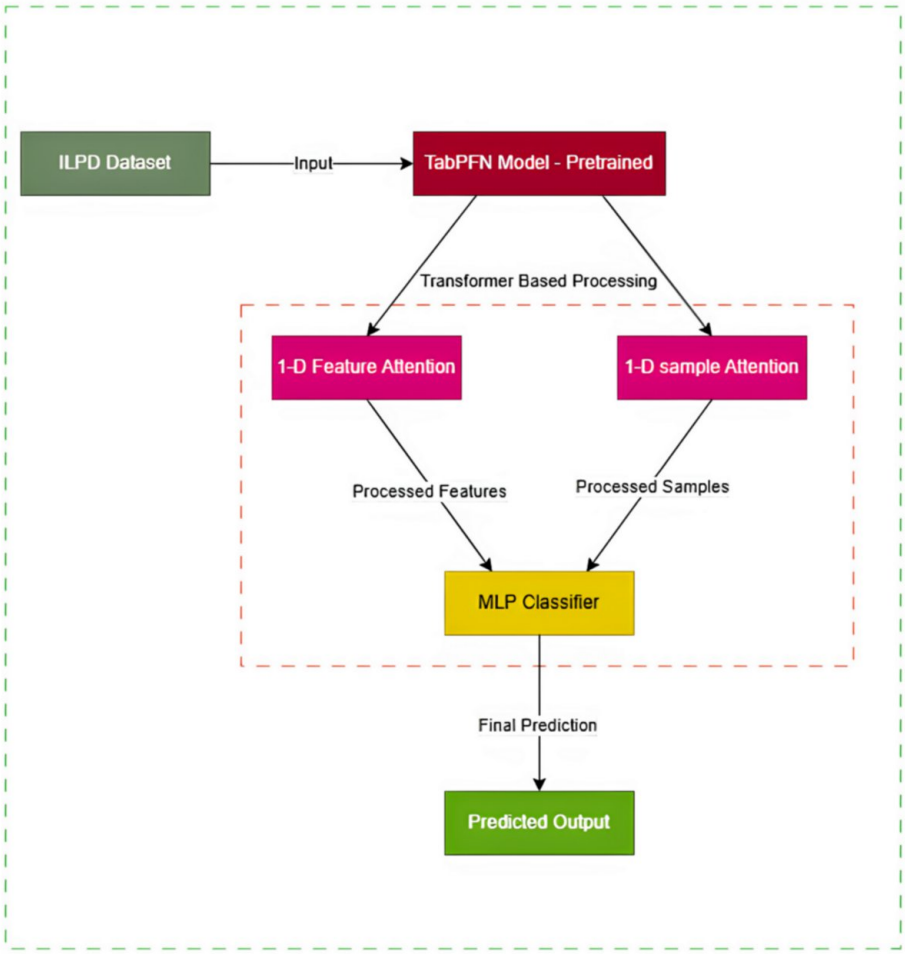

FIGURE 7: Illustration of the TabPFN model workflow, including transformer-based processing and prediction.

TabPFN, Prior-Data Fitted Network for Tabular Data; ILPD, Indian Liver Patient Dataset; MLP, Multilayer Perceptron

Results

The performance of the TabPFN classifier to forecast liver disease was evaluated by means of key performance metrics like accuracy, precision, recall, F1 measure, and AUC. The overall model performance for every class is provided in the classification report presented in Table *3*.

Class	Precision	Recall	F1-Score	Support
0 (No Disease)	0.86	0.96	0.91	80
1 (Disease Present)	0.96	0.85	0.90	86
Overall Accuracy			0.91	166
Macro Average	0.91	0.90	0.90	166
Weighted Average	0.91	0.90	0.90	166

TABLE 3: Classification report for the liver disease prediction.

As per the classification report, based on the weighted average, the model performs well in both classes, though class 1 has a negligible trade-off between precision and recall. Its high reliability in classifying instances of liver disease is attested by its overall weighted F1-score of 0.90.

Analysis of classification reports

The classifier correctly classified 90.36% of the instances in the test set, resulting in an overall accuracy of 90.36%. The model seems to be effective at handling class distributions according to balanced precision, recall, and F1-scores between the classes.

a) Accuracy: Accuracy is the proportion of correctly predicted cases out of all predictions. It is calculated as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

where FP and FN are false positives and false negatives, and TP and TN are true positive and true negative counts. The classifier has a good overall prediction quality with 90.36% accuracy.

b) Precision: It measures the number of accurately predicted positive cases divided by all projected positive cases. It is defined as follows:

$$Precision = \frac{TP}{TP + FP}$$

Fewer false positives are made by the model if its precision is good. In this case, precision is 0.86 for class 0 (healthy), which suggests that 86% of its no-disease predictions are correct, and 0.96 for class 1 (disease-present), which shows that 96% of its disease-present predictions are correct.

c) Recall: Recall measures the actual positive instances correctly detected by the model. It is defined as follows:

$$Recall = \frac{TP}{TP + FN}$$

A high recall shows that most actual positive instances are correctly identified by the model. For class 0, the recall of the classifier is 0.96, correctly identifying 96% of non-disease cases; for class 1, its recall is 0.85, correctly detecting 85% of actual disease cases.

d) F1-score: It is the harmonic mean of recall and precision that is particularly useful in scenarios when the classes have an unbalanced distribution. It is defined as follows:

$$F1\text{-}score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

A good balance between recall and precision is confirmed by a high F1-score. Its prediction performance is corroborated by both its 0.91 for class 0 and 0.90 F1-scores for class 1.

e) Macro and Weighted Averages: The macro average considers both classes uniformly and provides an unweighted mean of F1-score, precision, and recall. The weighted average ensures a balanced performance measure by considering the number of samples in a class.

Confusion matrix

By presenting the distribution of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN), the confusion matrix (Figure *8*) provides a handy representation of the model's performance in classification. Good predictive performance with minimal misclassification is shown by a dense concentration of values along the diagonal.

- TP = 73: The model correctly identified 73 cases of liver disease.
- TN = 77: The model correctly classified 77 non-diseased cases.
- FP = 3: The model misclassified three non-diseased cases as diseased (Type I error).
- FN = 13: The model failed to detect liver disease in 13 cases (Type II error).

The model's low FP and FN values show that it has a high ability to correctly classify both positive and negative cases. The classifier seems to be effective in detecting liver disease, as seen from the relatively low rate of FN, which reduces the likelihood of a missed diagnosis. Similarly, a low rate of unnecessary medical intervention is demonstrated by the relatively small number of FP.

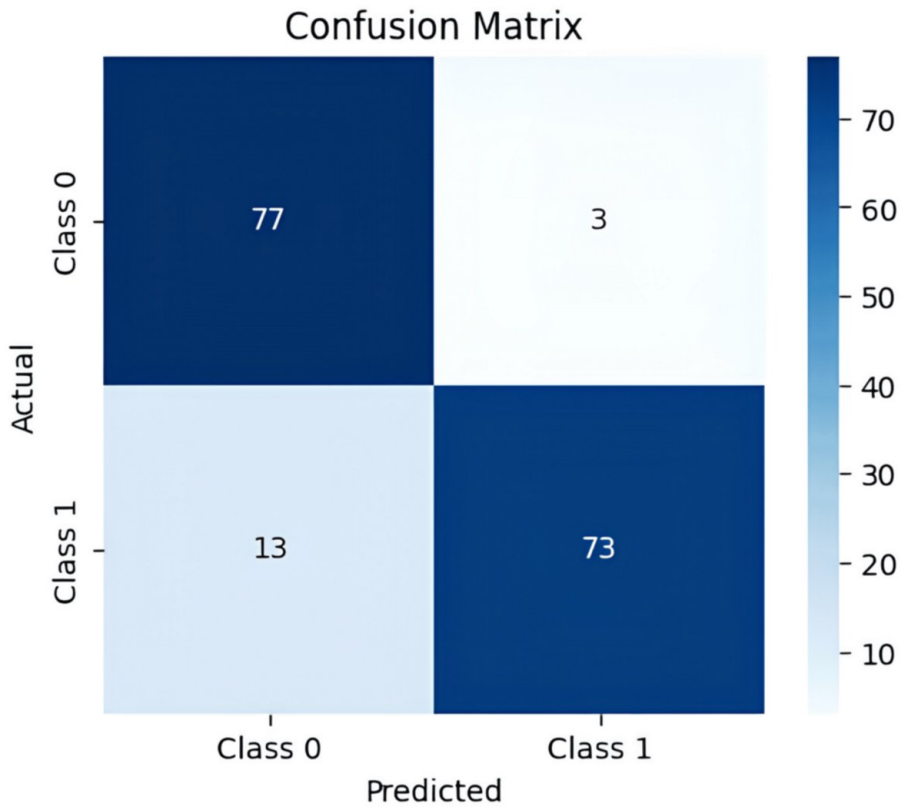

FIGURE 8: TabPFN classifier confusion matrix.

TabPFN, Prior-Data Fitted Network for Tabular Data

These results highlight the TabPFN classifier's overall good classification performance by demonstrating its dependability in differentiating between liver illness and its absence.

ROC curve

The false positive rate (FPR) and true positive rate (TPR) at various levels of categorization are shown in Figure *9*. The ability of the model to differentiate between negative and positive scenarios is illustrated through the ROC curve. The classifier's curve for a good-performing classifier approximately lies in the upper-left corner, indicating a high rate of TP and a low rate of FP. The model's high discriminatory ability is evidenced by the area under the ROC curve (AUC-ROC), which is 0.96. The classifier assigns true positive cases a greater likelihood than negative ones, as evidenced by an AUC value close to 1. The TabPFN classifier fares significantly better than a random guess, represented by the diagonal dashed line, which shows a random classifier with an AUC of 0.5. The AUC for the ROC is computed as follows:

$$AUC = \int_0^1 TPR \cdot d(FPR)$$

The greater classification ability is signaled by a higher AUC value, which identifies that the model effectively differentiates between liver diseases and non-liver diseases.

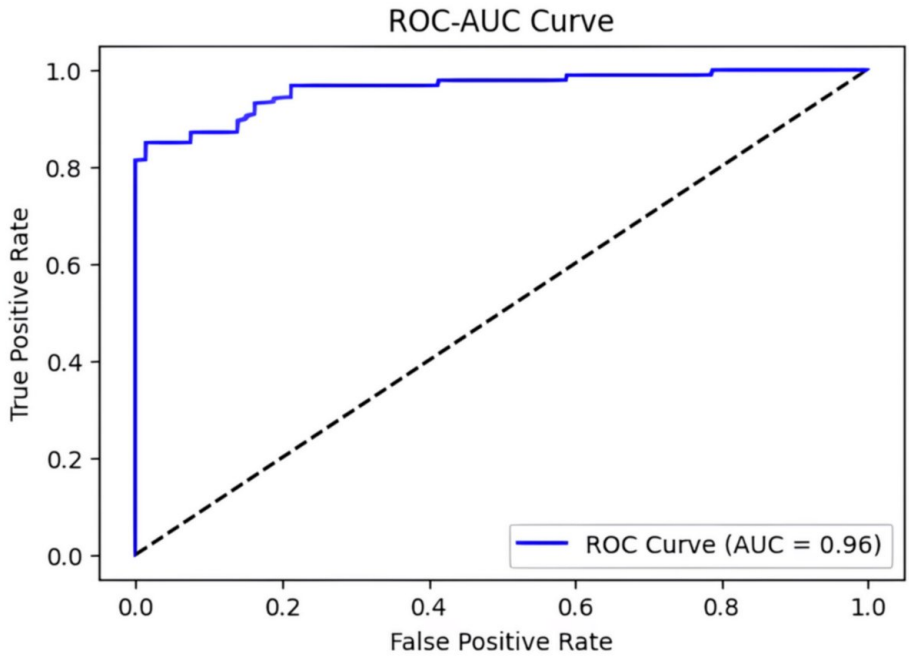

FIGURE 9: ROC-AUC curve (AUC = 0.96) demonstrating the classifier's discrimination ability.

AUC-ROC, Area Under the Receiver Operating Characteristic Curve

Precision-recall curve

The precision-recall (PR) curve is plotted to evaluate the trade-off between recall and precision, as presented in Figure *10*. The PR curve offers a clearer performance measure with precision plotted against recall at varying probability thresholds. This is specifically helpful for unbalanced datasets where one class occurs at a significantly higher frequency than the other.

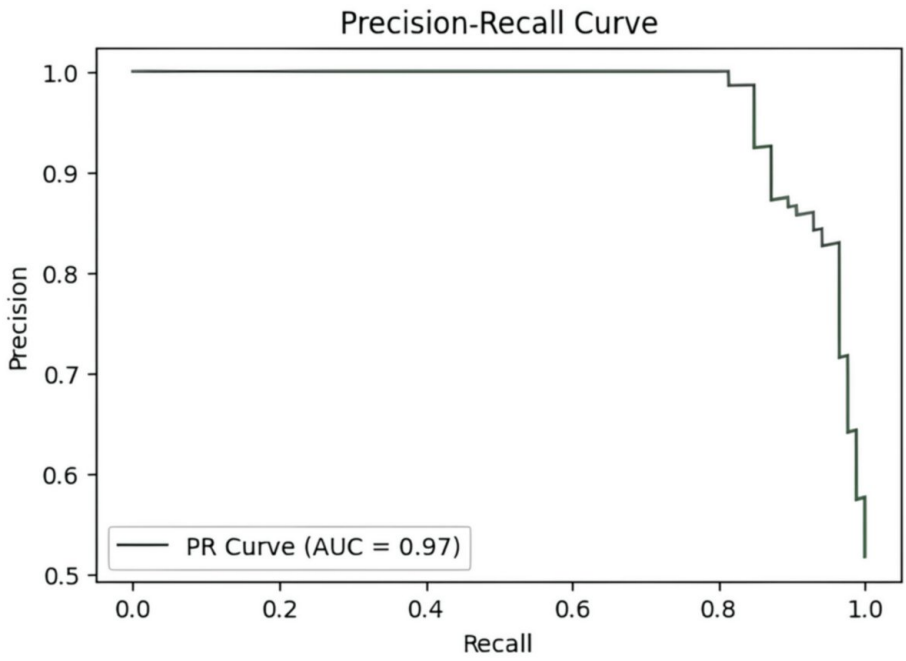

FIGURE 10: Precision-recall (PR) curve (AUC = 0.97), illustrating the balance between precision and recall.

AUC, Area Under the Curve

The PR-AUC value of 0.97 for the classifier's PR-AUC measure indicates that it has good recall but demonstrates strong precision. Recall (or sensitivity) measures the ability to pick up all actual positive cases, while precision indicates the proportion of correctly predicted positive cases out of all predicted positive cases. The classifier's strength in liver disease prediction is also verified by the high PR-AUC value, which indicates that it identifies positive cases with minimal FP with high efficiency.

In handling imbalanced distributions of data, PR-AUC provides more meaningful information compared to the ROC curve and is particularly helpful in medical diagnosis tasks. High PR-AUC reveals that the classifier is a reliable tool for disease prediction since it maintains high precision even at higher recall rates.

Discussion

This section evaluates the performance of the proposed TabPFN model in comparison to previous studies reported in the cited literature, as well as four widely used classification models. Accuracy, precision, recall, and F1-score are used to assess the performance of the models. The confusion matrix and ROC-AUC curve are also analyzed to provide a more comprehensive evaluation.

Comparison with previous studies

A performance benchmark is obtained from the results of earlier research using similar datasets. The classification performance of various models reported in earlier studies is given in Table *4*.

Model	Accuracy %	Precision %	Recall %	F1-Score %
LR [12]	75.25	78.43	91.95	72.06
SVM [12]	74.36	74.36	100.00	63.42
GBC [12]	71.79	78.12	86.21	70.00
DTC [12]	70.09	73.63	93.10	62.53
KNN [13]	70.46	74.54	89.13	81.18
RFC [13]	72.02	73.86	94.20	82.80
ANN [13]	71.50	71.87	100.00	83.38
TabPFN (proposed)	90.36	90.99	90.36	90.35

TABLE 4: Performance summary from literature study.

LR, Logistic Regression; SVM, Support Vector Machine; GBC, Gradient Boosting Classifier; DTC, Decision Tree Classifier; KNN, K-Nearest Neighbors; RFC, Random Forest Classifier; ANN, Artificial Neural Network; TabPFN, Prior-Data Fitted Network for Tabular Data

The results indicate that the proposed TabPFN model significantly outperforms traditional ML models. With an accuracy of 90.36%, it surpasses the highest accuracy achieved in prior studies, which remains below 76%. The model's performance in handling the classification tasks, reducing FP, and balancing sensitivity and specificity is also supplemented by the improvement in precision and F1-score. Although models such as ANN and SVM have a high recall score, their F1-scores are lower and indicate uneven precision, thereby rendering the classification system less reliable.

With its improved performance, TabPFN might be in a better position to generalize across different cases, minimize misclassification, and enhance decision-making regarding the detection of liver disease.

Performance evaluation on selected models

With the same dataset used for the TabPFN model, a comparison analysis is performed using four well-known classification models: RF, KNN, SVM, and LR. The results of training and testing these models are presented in Table *5*.

Model	Accuracy %	Precision %	Recall %	F1-Score %
Logistic Regression	66.26	64.15	79.06	70.83
Support Vector Machine	74.09	68.69	91.86	78.60
K-Nearest Neighbors	72.89	69.15	86.04	76.68
Random Forest	83.73	82.41	87.20	84.74
TabPFN (proposed)	**90.36**	**90.99**	90.36	**90.35**

TABLE 5: Performance comparison of selected classification models.

It is evident from Table *5* that the TabPFN model excels all traditional classification models on every key measure. Even in comparison with the best-performing traditional models, RF (83.73%) and SVM (74.09%), the accuracy rate of 90.36% is significantly higher, which is also shown in Figure *11*.

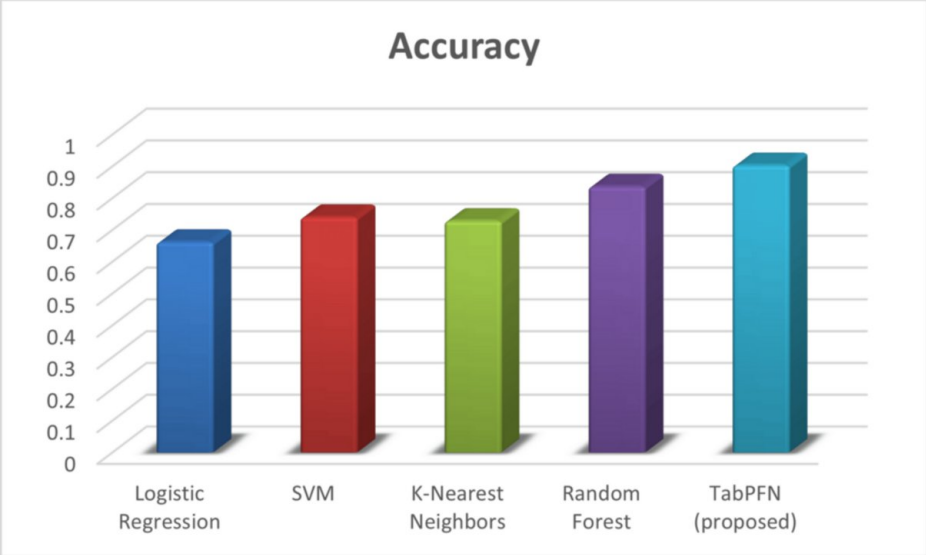

FIGURE 11: Performance comparison of classification models based on accuracy.

SVM, Support Vector Machine; TabPFN, Prior-Data Fitted Network for Tabular Data

Furthermore, the model's precision of 90.99% implies that it makes fewer FP predictions, which makes it particularly useful for medical diagnosis since FP could lead to unnecessary medical interventions. The visual comparison of different models based on precision is shown in Figure *12*.

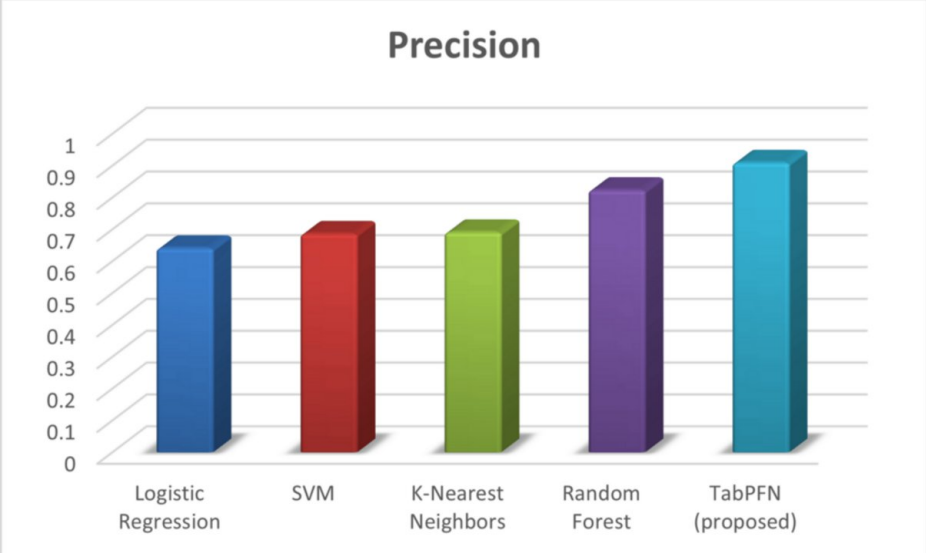

FIGURE 12: Performance comparison of classification models based on precision.

SVM, Support Vector Machine; TabPFN, Prior-Data Fitted Network for Tabular Data

SVM's lowest F1-score of 78.60%, while having its best recall of 91.86%, indicates that there is a compromise in which a significant proportion of FP sacrifice its overall reliability. Contrastingly, TabPFN has the highest F1-score of 90.35%, visually shown in Figure *13*, and has a high recall, which is an indication of a good balance between identifying TP cases and eliminating FPs.

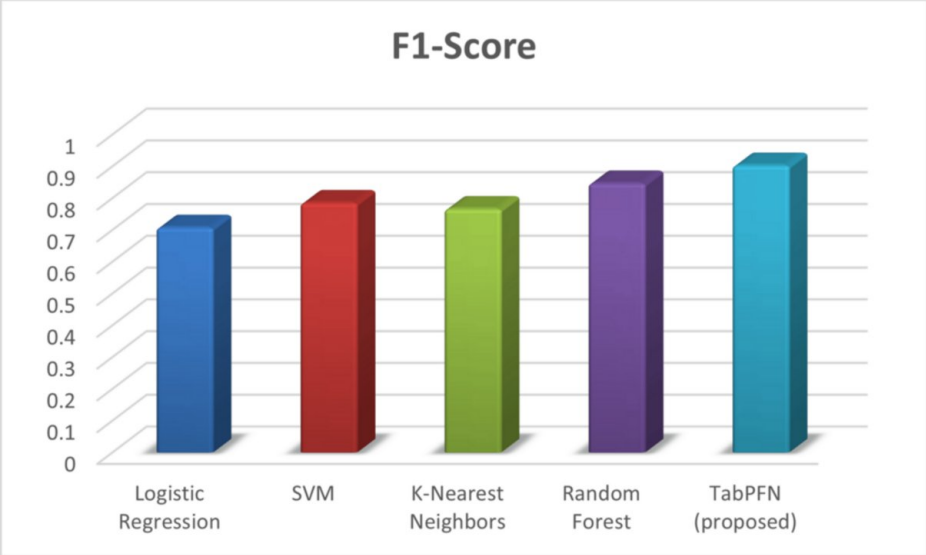

FIGURE 13: Performance comparison of classification models based on F1-score.

SVM, Support Vector Machine; TabPFN, Prior-Data Fitted Network for Tabular Data

Due to its balanced decision strategy and inherent PR trade-off, TabPFN was not besting all the models in terms of recall, also shown in Figure *14*. TabPFN keeps a tighter decision boundary, aiming for overall prediction performance, whereas SVM aims to maximize recall through detecting more items as positive (perhaps at the expense of greater FPs). Furthermore, with models such as SVM attempting to improve sensitivity for positive cases, the recall could be influenced by dataset characteristics and class distribution. Despite this, TabPFN is the most dependable and generalizable model because it has a good balance in all measures.

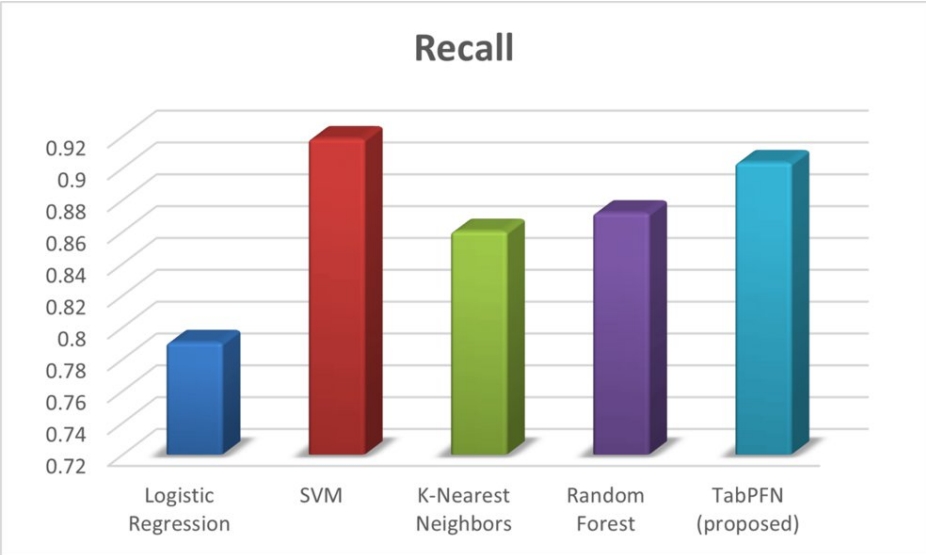

FIGURE 14: Performance comparison of classification models based on recall.

SVM, Support Vector Machine; TabPFN, Prior-Data Fitted Network for Tabular Data

Tables *4* and 5 show eloquent proof of the superiority of the TabPFN model as a classifier of liver diseases. The following points illustrate how effective it is:

• Higher Accuracy: The model has better generalization ability and an appreciable improvement over

traditional methods, with an accuracy of 90.36%.

• Balanced Precision and Recall: TabPFN achieves robust classification with a balance between precision and recall, while other models generally prioritize one at the expense of the other.

• Superior F1-Score: The fact that the model handles class imbalances, an essential part of medical diagnosis where minimizing FPs as well as FNs is important, is established through the F1-score of 90.35%.

• Reduced Misclassification: The confusion matrix analysis gives stronger support to the hypothesis that TabPFN accurately discriminates between disease-positive and disease-negative cases with fewer misclassifications.

• Robustness Against Class Imbalance: The class imbalance robustness of the model is supported by the PR-AUC and ROC-AUC measures, which in medical datasets, especially when class distributions are unbalanced, prove beneficial.

All in all, the TabPFN model demonstrates state-of-the-art classification performance and is an extremely effective approach to forecasting liver disease. Its practicality for applications in the medical profession is emphasized through its high precision, recall, F1-score, and accuracy.

ROC-AUC curve analysis

A complete analysis of the trade-off between TPR and FPR for different categorization criteria is provided by the ROC curve. One key measure for assessing the discrimination abilities of a model is the AUC. Better class distinction ability is shown by a larger AUC. Figure *15* presents the ROC-AUC curves for every model.

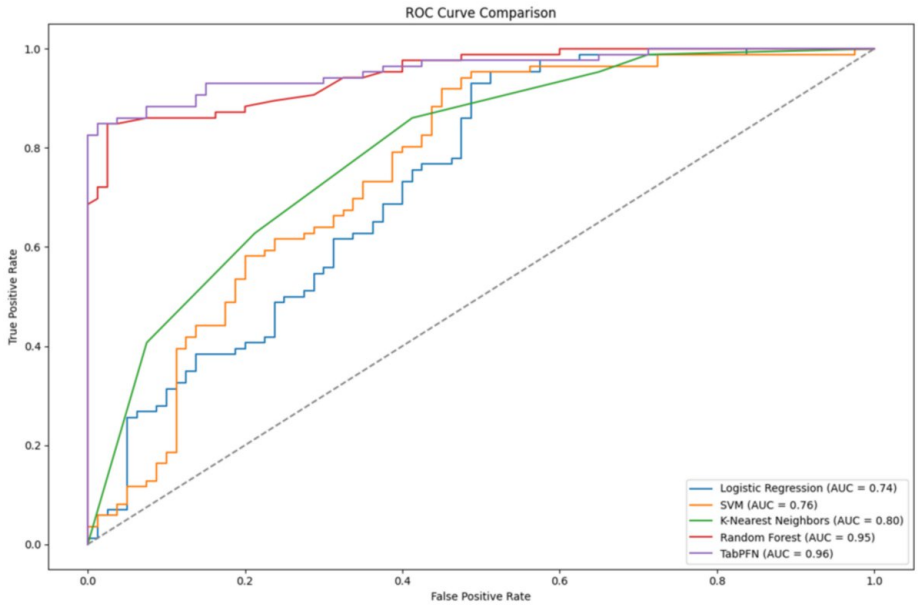

FIGURE 15: ROC-AUC curve for different models.

AUC-ROC, Area Under the Receiver Operating Characteristic Curve; SVM, Support Vector Machine

TabPFN is the best among all models in class differentiation, as indicated by its highest AUC value of 0.96. By keeping the TPR high and FPs low, TabPFN enhances the reliability of classification.

Comparison With Other Models

• RF (AUC = 0.95): With a very good classification power, it performs in very close proximity to TabPFN. Its ROC curve, however, depicts slightly lesser robustness at low FPRs compared to TabPFN.
• KNN (AUC = 0.80): While having a fairly good performance, it contains greater FPRs, thus yielding less than optimal class discriminating.
• SVM (AUC = 0.76) and LR (AUC = 0.74): They show lower AUC values, reflecting their limitations in maintaining a balance between sensitivity and specificity.

Based on the findings, TabPFN has the optimal class separation, which indicates that it has good generalization under a variety of decision criteria. Its performance is due to its transformer-based processing, which enhances feature representation and model flexibility. It is a better choice for disease detection applications because of its improved AUC score, which again justifies its ability to cope with difficult classification tasks.

Analysis of confusion matrix

The confusion matrices for all models are presented in Figure *16*. The matrices provide insights into the models' misclassification tendency and the ability to classify examples correctly.

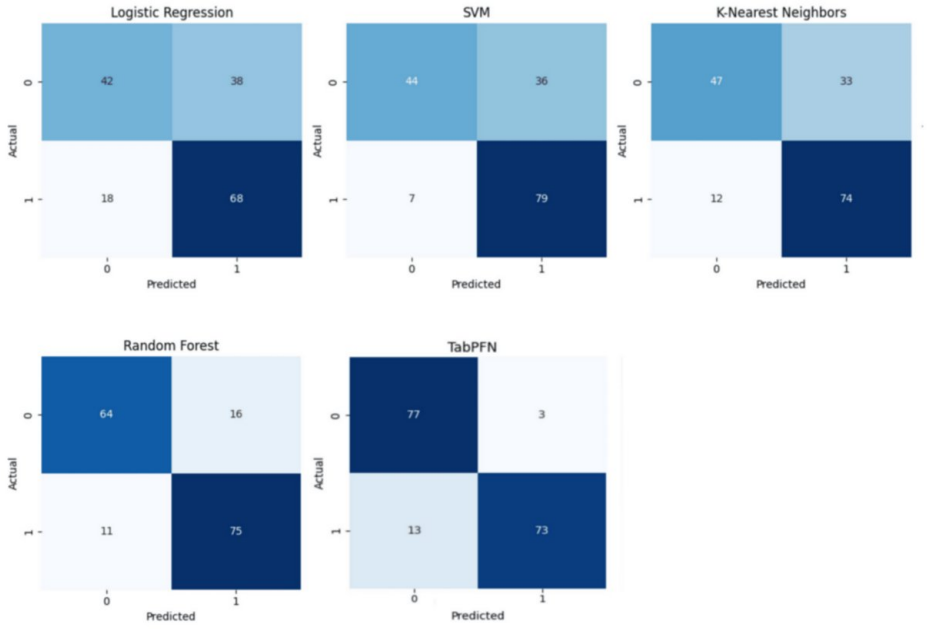

FIGURE 16: Confusion matrices for different models.

SVM, Support Vector Machine; TabPFN, Prior-Data Fitted Network for Tabular Data

Regarding the differentiation of the two classes, TabPFN achieves the minimum misclassification error, as is seen in its confusion matrix. There are 16 misclassified cases in total (13 FNs and 3 FPs) that the model misses,
but it successfully identifies 77 TNs and 73 TPs. This conforms to the results in Table *2*, which reflect its outstanding precision and accuracy.

Comparison With Other Models

• SVM identifies 79 TPs with the best recall. However, this lowers precision by raising FPs (36).
• LR, though works relatively well, also carries a large number of FPs (38), along with FNs (18), which decreases its overall predictive reliability.
• KNN, by correctly identifying 74 TPs, is better than LR at recall; however, it also retains 33 FPs.
• RF, with 64 TNs and 75 TPs, both classified correctly and relatively fewer misclassifications (16 FPs and 11 FNs), does better than KNN and SVM concerning balance.

The confusion matrix of TabPFN illustrates its excellent decision-making balance, guaranteeing excellent generalization by limiting FPs and FNs. The optimal overall F1-score is achieved by maintaining a robust recall while limiting misclassification, particularly FPs, which results in higher precision.

Summary of findings

According to numerous significant measures of evaluation, experimental results prove that TabPFN always surpasses conventional ML algorithms.

• TabPFN proved its overall supremacy in classification by earning the highest accuracy, precision, and F1-score.
• It effectively minimizes FPs and FNs by achieving a balance between recall and precision.

• It minimizes classification bias to the majority class and performs well on imbalanced datasets.
• It is the most reliable classifier among the models tested, as indicated by its excellent generalization capability and highest AUC score of 0.96.

These findings affirm TabPFN's promise as the highest-performing model, particularly for applications where high recall, accuracy, and precision are essential.

Overall, TabPFN possesses state-of-the-art categorization features, making it a strong contender for future ML applications.

Conclusions

This paper illustrates the effectiveness of TabPFN in liver disease classification, especially on small and unbalanced datasets. With an ROC-AUC of 0.96, accuracy of 90.36%, precision of 90.99%, and F1-score of 90.35%, TabPFN outperformed traditional models such as KNN, RF, LR, and SVM when evaluated on ILPD. Without needing extensive feature engineering or hyperparameter optimization, TabPFN is able to learn complex feature interactions due to its transformer-based architecture. Due to this, it is especially useful for medical applications where resources and data are limited. Its practical application is evidenced by its superior minority class prediction performance, as illustrated by ROC curves and confusion matrices.

Unlike the conventional models that require a lot of human effort, TabPFN offers a plug-and-play approach that simplifies the development of diagnostic tools. To ensure its generalizability and enhance interpretability for therapeutic application, further studies on larger and more diverse datasets are needed. In summary, TabPFN provides a powerful, accurate, and efficient way to predict liver disease and has immense potential for integration into clinical decision support systems based on AI.

Additional Information

Author Contributions

All authors have reviewed the final version to be published and agreed to be accountable for all aspects of the work.

Concept and design: Pradeep Rawat

Acquisition, analysis, or interpretation of data: Pradeep Rawat, Prabh Deep Singh, Vikas Tripathi

Drafting of the manuscript: Pradeep Rawat

Critical review of the manuscript for important intellectual content: Pradeep Rawat, Prabh Deep Singh, Vikas Tripathi

Supervision: Prabh Deep Singh, Vikas Tripathi

Disclosures

Human subjects: All authors have confirmed that this study did not involve human participants or tissue. **Animal subjects:** All authors have confirmed that this study did not involve animal subjects or tissue. **Conflicts of interest:** In compliance with the ICMJE uniform disclosure form, all authors declare the following: **Payment/services info:** All authors have declared that no financial support was received from any organization for the submitted work. **Financial relationships:** All authors have declared that they have no financial relationships at present or within the previous three years with any organizations that might have an interest in the submitted work. **Other relationships:** All authors have declared that there are no other relationships or activities that could appear to have influenced the submitted work.

References

1. Devarbhavi H, Asrani SK, Arab JP, Nartey YA, Pose E, Kamath PS: Global burden of liver disease: 2023 update. Journal of Hepatology. 2023, 79:516-537. 10.1016/j.jhep.2023.03.017
2. Vento S, Cainelli F: Chronic liver diseases must be reduced worldwide: it is time to act. The Lancet Global Health. 2022, 10:e471-e472. 10.1016/s2214-109x(22)00047-x
3. Afreen N, Patel R, Ahmed M, Sameer M: A novel machine learning approach using boosting algorithm for liver disease classification. 2021 5th International Conference on Information Systems and Computer Networks (ISCON). 2021, 1-5. 10.1109/ISCON52037.2021.9702488
4. Ardchir S, Ouassit Y, Ounacer S, Ghoumari MYE, EL Azzouazi M: An integrated ensemble learning framework for predicting liver disease. International Journal of Online and Biomedical Engineering (iJOE). 2023, 19:138-152. 10.3991/ijoe.v19i13.41871
5. Ramana B, Venkateswarlu N: ILPD (Indian Liver Patient Dataset) [Dataset]. UCI Machine Learning Repository.

(2022). https://doi.org/10.24432/C5D02C.
6. Arık SÖ, Pfister T: TabNet: Attentive interpretable tabular learning. Proceedings of the AAAI Conference on Artificial Intelligence. 2021, 35:6679-6687. 10.1609/aaai.v35i8.16826
7. Hollmann N, Müller S, Eggensperger K, Hutter F: TabPFN: A transformer that solves small tabular classification problems in a second. arXiv:2207.01848. 10.48550/arXiv.2207.01848
8. Hollmann N, Müller S, Purucker L, et al.: Accurate predictions on small data with a tabular foundation model. Nature. 2025, 637:319-326. 10.1038/s41586-024-08328-6
9. Srivastava A, Vineeth Kumar V, Mahesh TR, Vivek V: Automated prediction of liver disease using machine learning (ML) algorithms. 2022 Second International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT). 2022, 1-4. 10.1109/ICAECT54875.2022.9808059
10. Gamara RPC, Teologo AT, Neyra RQ, Bandala AA: AI-based diagnostic tool for liver disease using machine learning algorithms. 2022 IEEE 14th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM). 2022, 1-6. 10.1109/hnicem57413.2022.10109367
11. Yadav P, Sharma SC, Yadav H: Deep learning based framework for multi-disease detection using CNN-BiLSTM. Proceedings of the 12th International Conference on Soft Computing for Problem Solving. Pant M, Deep K, Nagar A (ed): Springer, Singapore; 2024. 995:693-706. 10.1007/978-981-97-3292-0_47
12. Mohamed S, Ezzat R, Ghorab S, Bhatnagar R, Shams MY: Liver disease identification based on machine learning algorithms. 2023 3rd International Conference on Technological Advancements in Computational Sciences (ICTACS). 2023, 1062-1067. 10.1109/ictacs59847.2023.10390061
13. Shivaji Rao SS, Gangadhara Rao K: Diagnosis of liver disease using ANN and ML algorithms with hyperparameter tuning. 2024 2nd International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT). 2024, 629-634. 10.1109/IDCIoT59759.2024.10467855
14. Minnoor M, Baths V: Liver disease diagnosis using machine learning. 2022 IEEE World Conference on Applied Intelligence and Computing (AIC). 2022, 41-47. 10.1109/aic55036.2022.9848916
15. Anthonysamy V, Khadar Babu SK: Multi perceptron neural network and voting classifier for liver disease dataset. IEEE Access. 2023, 11:102149-102156. 10.1109/access.2023.3316515
16. Kumar S, Rani P: Liver disease prediction using Bayesian optimized classification algorithms. 2024 2nd World Conference on Communication & Computing (WCONF). 2024, 1-6. 10.1109/wconf61366.2024.10692281
17. Jaya Mabel Rani A: Liver disease prediction using hybrid support vector ma chine and fuzzy centroid based clustering algorithm. 2023 International Conference on Data Science, Agents & Artificial Intelligence (ICDSAAI). 2023, 1-6. 10.1109/icdsaai59313.2023.10452431
18. Riya, Kaur B: Liver disease prediction using machine learning algorithms. International Journal of Computer Applications. 2023, 185:36-44. 10.5120/ijca2023923022
19. Zhao R, Wen X, Pang H, Ma Z: Liver disease prediction using W-LR-XGB algorithm. 2021 International Conference on Computer, Blockchain and Financial Development (CBFD). 2021, 245-248. 10.1109/cbfd52659.2021.00055
20. Amin R, Yasmin R, Ruhi S, Rahman MH, Reza MS: Prediction of chronic liver disease patients using integrated projection based statistical feature extraction with machine learning algorithms. Informatics in Medicine Unlocked. 2023, 36:101155. 10.1016/j.imu.2022.101155