Computer-Aided Diagnosis of Complications after Liver Transplantation Based on Transfer Learning

Abstract

Liver transplantation is one of the most effective treatments for acute liver failure, cirrhosis, and even liver cancer. The prediction of postoperative complications is of great significance for liver transplantation. However, the existing prediction methods based on traditional machine learning are often unavailable or unreliable due to the insufficient amount of real liver transplantation data. Therefore, we propose a new framework to increase the accuracy of computer-aided diagnosis of complications after liver transplantation with transfer learning, which can handle small-scale but high-dimensional data problems. Furthermore, since data samples are often high-dimensional in the real world, capturing key features that influence postoperative complications can help make the correct diagnosis for patients. So, we also introduce the SHapley Additive exPlanation (SHAP) method into our framework for exploring the key features of postoperative complications. We used data obtained from 425 patients with 456 features in our experiments. Experimental results show that our approach outperforms all compared baseline methods in predicting postoperative complications. In our work, the average precision, the mean recall, and the mean F1 score reach 91.22%, 91.70%, and 91.18%, respectively.

Keywords: liver transplantation, prediction of complications, transfer learning, feature importance

1 Introduction

In the past decades, liver transplantation has become a routine clinical operation and plays a significant role in the clinical treatment of liver cancer [1]. However, the probability of postoperative complications cannot be ignored, such as primary graft non-function (PNF), postoperative bleeding, wound infection, and bile leakage. These complications have an incidence rate of about 14% to 35%, which is an important risk factor affecting the quality of life and survival time of the recipients [2]. For instance, PNF is one of the most severe postoperative complications after liver transplantation. It often endangers the recipient's life and leads to graft failure [3]. Although the incidence of PNF is not high compared to other complications, it ranges from 2% to 6% [4]. Once PNF occurs, a second liver transplantation should be performed; otherwise, the survival rate is almost 0. Postoperative bleeding is also a common complication after liver transplantation, which can occur at any time after the surgery, usually within 48 hours. The incidence rate of bile duct complications, ranging between 5% and 50%, often affects the quality of long-term life of recipients and leads to liver transplantation failure and poor survival [5].

Therefore, successful prediction and proper treatment of postoperative complications are of great significance for liver transplantation. Currently, the detection of liver transplantation complications still relies on manual screening and regular review [6]. This is mainly due to the small dataset of liver transplantation and the large feature space, which is not suitable for machine learning algorithms [7, 8]. In traditional machine-learning-based prediction models, the diagnosis of liver transplantation complications is considered as a statistical classification problem using clinical manifestations as features. First, a feature space is created based on clinical manifestation. Then, a statistical model is used to train the classification function [9]. However, the prediction accuracies of such classic models heavily depend on the number of available liver transplantation cases. It is difficult for mainstream machine learning algorithms to learn knowledge from a few samples with a large number of features. Furthermore, most deep learning models suffer from insufficient interpretability. Therefore, it is challenging to identify the key features that contribute to the prediction of postoperative complications. This limits the application of deep learning models in clinical practice.

In this paper, we propose a Transfer Component Analysis (TCA)-based prediction method for postoperative complications of liver transplantation. Given the small sample size and large feature space, we utilize feature-based transfer learning methods such as TCA to adaptively select the best latent feature space, which helps to reduce the feature space and save training time. Additionally, we introduce the Random Over-Sampling (ROS) approach to dealing with imbalanced positive and negative samples. For the complication classification task, we adopt the support vector machine (SVM) after comprehensive comparisons with other models. To address the problem of insufficient interpretability in current complication prediction methods, we introduce the SHapley Additive exPlanation (SHAP) method into our framework [10]. The combined model of TCA and SVM is then interpreted together by SHAP. Accordingly, importance weights for key features of postoperative complications are generated.

The main contributions of this paper include:

1. A transfer learning-based prediction method is proposed for postoperative complications of liver transplantation. It can adaptively select an optimal latent feature space, which helps to reduce the feature space dimension and improve the model predictive performance.
2. It divides the dataset into the source and target domains. Both consist of data samples containing all three periods, namely, pre-operation, intra-operation, and post-operation. The method effectively enlarges the dataset and addresses the common problems related to small-sized datasets that have a large feature space.
3. We combine TCA and SVM together and utilize SHAP to interpret the combined model, solving the issue of insufficient interpretability in the current prediction models.
4. The importance weights for original features are provided to aid in the analysis and diagnosis of postoperative complications of liver transplantation.

The rest of this paper is organized as follows. Related work is described in Section 2. Section 3 introduces a transfer learning-based computer-aided method for predicting postoperative complications of liver transplantation. The evaluation of the proposed approach is given in Section 4. Section 5 concludes our work with final remarks.

2 Related Work

Prognosis evaluation is the prediction of potential outcomes for a patient after being diagnosed with a disease. This is extremely important in the field of clinical medicine. As different patients can respond differently to the same treatment plan, due to factors such as age, overall health, and the presence of other illnesses, it is crucial to accurately obtain a patient's prognosis in order to create targeted treatment plans. This can significantly improve patients' survival rates.

Asadi et al. [11] employed artificial neural networks (ANN) and support vector machines (SVM) for prediction of the final outcomes of brain arteriovenous malformations (BAVM) in patients who received endovascular technique treatment. The prediction accuracy was higher than 90%. They also analyzed the proportion of various factors that affect the effect of BAVM endovascular embolization. Similarly, Hope et al. [12] trained a Gaussian process regression model using magnetic resonance imaging (MRI) images to explore the causal relationship between lesions and treatment outcomes. Through this model, they predicted the severity of post-stroke cognitive impairment and its subsequent rehabilitation process over time. The aforementioned research findings demonstrate the effectiveness of machine learning in the assessment of disease marginal.

The existing prediction models for complications of liver transplantation are mainly built using logistic regression methods, with predictive variables typically derived from the clinical and pathological characteristics of liver transplantation [13]. However, these models often struggle to be both efficient and reliable due to the limited number of liver transplantation cases available. This small sample size makes it challenging for mainstream machine learning algorithms to learn effectively. Additionally, existing deep learning models often lack interpretability, making it difficult to understand how they arrived at their predictions.

Currently, the main methods to address the problem of small sample size include transfer learning and sampling techniques [14]. Feature-based transfer learning involves transforming data extracted from both the source and target domains into a new feature space. The data distributions obtained from these two domains are similar. By doing this, the labeled data from both domains can be used to train models on the new feature space, and the data from the target domain can be used to test the models [15]. This approach allows the use of knowledge learned from the source domain to improve the model performance in the target domain. Sampling techniques, on the other hand, involve generating a balanced training set from the original sample set, which can improve performance, particularly in terms of recall. However, it's important to note that the use of sampling may lead to an adverse impact on the overall effectiveness of the model if the sample imbalance is not severe.

Many machine learning algorithms lack interpretability, which makes it difficult to understand how they arrive at their predictions. To improve interpretability, there are two main approaches: using interpretable models or using model-independent interpretation tools. Interpretable models include decision trees, decision rules, and linear regression, among others. Model-independent interpretation tools, such as partial dependence plots, Local surrogate models (LIME), and SHapley Additive exPlanations (SHAP), provide explanations for the predictions made by machine learning models.

Transfer learning also has limitations in terms of interpretability. The process by which knowledge is transferred during learning requires further experimentation and theoretical verification. Recently, researchers from the University of Sydney, Australia, published their findings at IJCAI 2017 to better understand how features are transferred [16]. They theoretically proved that feature structures can be transferred independently of changes in conditional distributions between domains, thereby improving the performance of models in the target domain. However, their approach did not provide a method for calculating the explicit mapping relationship between the features in the original space and the latent space during the transfer process. In contrast, our approach directly obtains the original feature importance by interpreting the combined model without calculating the mapping relationship, thus reducing the impact of the transfer learning process on interpretability.

3 Method

As depicted in Fig. 1, we propose a framework for predicting postoperative complications of liver transplantation. The framework consists of three main modules. The first module is a feature-based transfer learning module, which maps features from the source and target domains to the same latent space. The second module is designed to handle imbalanced positive and negative samples. It uses the Random Over-sampling method, and then applies a machine learning-based method such as SVM to predict postoperative complications using the generated dataset from the latent space. The last module is the interpretation module, which interprets the combined model of TCA and SVM and accordingly, obtains the feature importance of the original data samples.

3.1 Feature-based Transfer Learning Module

Transfer learning is a technique aims to transfer knowledge across tasks or domains which are different but related [17–19]. One of the key goals of transfer learning is to overcome the problem of limited labeled samples available in the target domain [20]. It relaxes a fundamental assumption that most of the traditional machine learning algorithms make, which assumes that the training and testing data are identical in both the feature space and the probability distribution.

In transfer learning, a domain $\mathcal{D}$ is considered as composing of the feature space $\mathcal{X}$ of the input instance $X = \{x_1, \cdots, x_n\}$ and the marginal probability distribution of inputs $P(X)$, namely, $\mathcal{D} = \{\mathcal{X}, P(X)\}$. Marginal probabilities often vary across different domains. The task $\mathcal{T}$ consists of the label space $Y = \{y_1, \cdots y_n\}$ and the predictive function $f(\cdot)$, i.e., $\mathcal{T} = \{Y, f(\cdot)\}$. The problem of transfer learning can be defined as: given a source domain $\mathcal{D}_S$ and a learning task $\mathcal{T}_S$, as well as a target domain

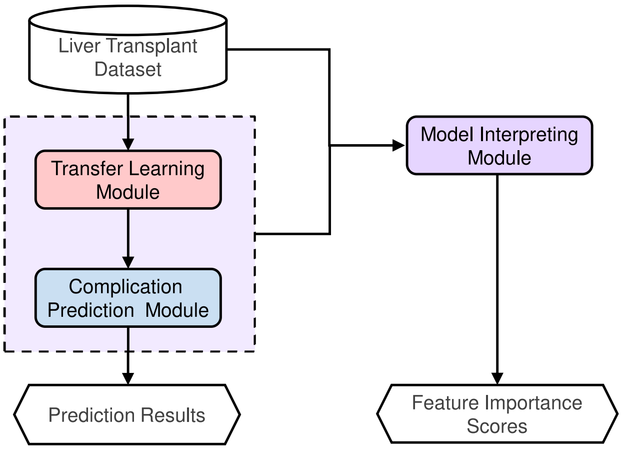

Liver Transplant Dataset
Transfer Learning Module
Complication Prediction Module
Model Interpreting Module
Prediction Results
Feature Importance Scores

Fig. 1: The proposed framework for complication prediction.

$\mathcal{D}_T$ and a learning task $\mathcal{T}_T$ (where $\mathcal{D}_T$ is different from D_S), the task is to obtain the proper predicting function $f(\cdot)$ for $\mathcal{D}_T$ by leveraging the knowledge acquired from the source domain [15].

Due to the limited number of real-life liver transplantation cases and the high-dimensional feature space, we cannot use traditional machine learning algorithms to predict postoperative complications directly. To overcome this challenge, we use transfer learning to enhance the dataset and reduce feature dimensionality. When reducing the number of features, feature selection methods are commonly employed, such as statistical analysis, clinical experience, or the Least Absolute Shrinkage and Selection Operator (LASSO). These methods select m features from the original n features, and the remaining $(n - m)$ features are discarded. However, transfer learning, such as TCA, can use m new features to represent the information of the original n features ($m < n$). It projects the original n features of the data into an m-dimensional latent space while keeping the two domain distributions close and maximizing the data variance. Compared to feature selection, transfer learning retains more information contained in the original features, which enhances the predictive performance of the model. Transfer learning approaches also facilitate the interpreting module in the proposed model to find the risk factors for complications because it retains more components of the original features in the input to the downstream prediction model.

Transfer learning encompasses two primary approaches commonly employed. One approach is instance-based transfer learning, also known as the instance weighting method. This method assumes that the marginal distribution of the target domain $P(X_T)$ can be supported by the marginal distribution of the source domain $P(X_S)$, i.e., $P(X_T)/P(X_S) < \infty$, and that the conditional distributions of the two domains $P(Y_S \mid X_S) = P(Y_T \mid X_T)$ are identical. However, this approach is only valid when the distribution difference between the two domains is small [21]. The other prevalent approach is known as feature-based transfer learning, which operates under the premise that there is a shared and generalizable latent feature space between the two domains.

This latent feature space can be obtained by feature learning algorithms aiming to mitigate the distribution disparities between the domains [15]. This approach can be used in cases where there are large differences between domain distributions. In this work, our main emphasis is on feature-based transfer learning. In our partitioned dataset, the source and target domains use the values of the same medical index at different moments. Therefore, the conditional distributions $P(Y_S \mid X_S)$ and $P(Y_T \mid X_T)$ are different, but the feature spaces remain similar between the two domains. This makes it difficult to handle using the instance-based method.

Feature-based transfer learning can be categorized into homogeneous and heterogeneous styles, with Transfer Component Analysis (TCA) and Progressive Alignment Heterogeneous Domain Adaptation (HDA) being the most representative methods, respectively. Homogeneous transfer learning aims to bridge the gap that exists between the domain distributions [22] and is used when the source and target domains possess analogous feature spaces, i.e., $\mathcal{X}_S$ and $\mathcal{X}_T$ are comparable. In this work, we adopt TCA as the main module in the proposed framework, as it falls under homogeneous transfer learning that projects the original features of the two domains into a shared latent space. This helps to reduce the domain shift between them. HDA, on the other hand, is used as a baseline method for comparison.

3.1.1 Transfer Component Analysis (TCA)

An effective method of homogeneous transfer learning is Transfer Component Analysis (TCA). TCA transforms the features into a same latent space, thus reducing the disparities in probability distributions between the source and target domains. This allows traditional machine learning methods to be used for classification prediction [17].

$$\operatorname{Dist}\left(X_S', X_T'\right) = \left\| \frac{1}{n_T} \sum_{i=1}^{n_T} \left(x_{T_i}\right) - \frac{1}{n_S} \sum_{i=1}^{n_S} \left(x_{S_i}\right) \right\|_{\mathcal{H}}^2 \tag{1}$$

TCA relies on the assumption that $P\left(X_S\right) \neq P\left(X_T\right)$, and it is possible to achieve $P\left(\phi\left(X_S\right)\right) \approx P\left(\phi\left(X_T\right)\right)$ and $P\left(Y_S \mid \phi\left(X_S\right)\right) \approx P\left(Y_T \mid \phi\left(X_T\right)\right)$ through a transformation ϕ, in which X_S and X_T represent inputs from the source and target domains. The transformation ϕ is assumed as the feature mapping function. ϕ can be found by minimizing the Maximum Mean Discrepancy (MMD) between domains as indicated by formula (1) to increase the similarity between domains after transformation, where $x_{S_i} \in X_S$, $x_{T_i} \in X_T$, $X_S' = \phi\left(X_S\right)$, and $X_T' = \phi\left(X_T\right)$. However, it is hard to find such a transformation ϕ because of the highly nonlinear nature of ϕ, and explicit optimization of formula (1) may result in suboptimal local minima. To circumvent this challenge, Pan et al [17] transformed the problem into a kernel learning problem rather than obtain it directly. The kernel trick $\left(k\left(x_i, x_j\right) = \phi\left(x_i\right)' \phi\left(x_j\right)\right)$ allows us to describe the MMD between the two domains in formula (1) as formula (2).

$$\text{Dist}\left(X_S', X_T'\right) = \operatorname{tr}\left(\widetilde{K}L\right) \tag{2}$$

where

$$\widetilde{K}K = \begin{bmatrix} K_{S,S} & K_{S,T} \\ K_{T,S} & K_{T,T} \end{bmatrix} \tag{3}$$

$$L_{ij} = \begin{cases} \frac{1}{n_S^2} & x_i, x_j \in X_S \\ \frac{1}{n_T^2} & x_i, x_j \in X_T \\ -\frac{1}{n_S n_T} & \text{otherwise} \end{cases} \tag{4}$$

$K_{S,S}$, $K_{S,T}$, and $K_{T,T}$ respectively are the Gram matrices of the source, cross, and target domain data in the embedded space. $\widetilde{K}$ is a $(n_1 + n_2) \times (n_1 + n_2)$ matrix, where n_1 and n_2 indicate the size of the source and target domains. The kernel $k(\cdot, \cdot)$ can be calculated utilizing an algorithm known as Maximum Mean Discrepancy Embedding (MMDE) [23, 24]. Due to the limitations of MMDE, Pan et al. [17] presented a unified kernel learning method called Transfer Component Analysis (TCA), which uses an explicit low-rank representation. In TCA, the $(n_1 + n_2) \times m$ matrix W is used to map the samples to a latent representation space with dimension m, where $m \ll n_1 + n_2$, which also reduces the dimensionality of features. The corresponding transformation of formula (2) is written as formula (5).

$$\text{Dist}\left(X_S', X_T'\right) = \operatorname{tr}\left(W^\top \widetilde{K} L \widetilde{K} W\right) \tag{5}$$

The distance minimizing problem then reduces to formula (6).

$$\begin{aligned} &\min_W \operatorname{tr}\left(W^\top \widetilde{K} L \widetilde{K} W\right) + \lambda \operatorname{tr}\left(W^\top W\right) \\ &\text{s.t. } W^\top \widetilde{K} H \widetilde{K} W = I_m \end{aligned} \tag{6}$$

where $\operatorname{tr}\left(W^\top W\right)$ is employed to regulate the complexity of W. λ serves as a hyper-parameter representing the weight. $I_m \in \mathbb{R}^{m \times m}$ represents the identity matrix, and m denotes the latent space dimension. As described in formula (7), H is a centering matrix and $\mathbf{1}$ represents an all-1 vector of $n_1 + n_2$ dimensions. Besides, the constraint $W^\top \widetilde{K} H \widetilde{K} W = I_m$ is proposed to preclude the $W = 0$ trivial solution.

$$H = I_{n_1+n_2} - \frac{1}{n_1 + n_2} \mathbf{1}\mathbf{1}^\top \tag{7}$$

It has been proved in [17] that the problem in (6) can be solved by trace optimization. The solution of W in (6) consists of the latent vectors that correspond to the m

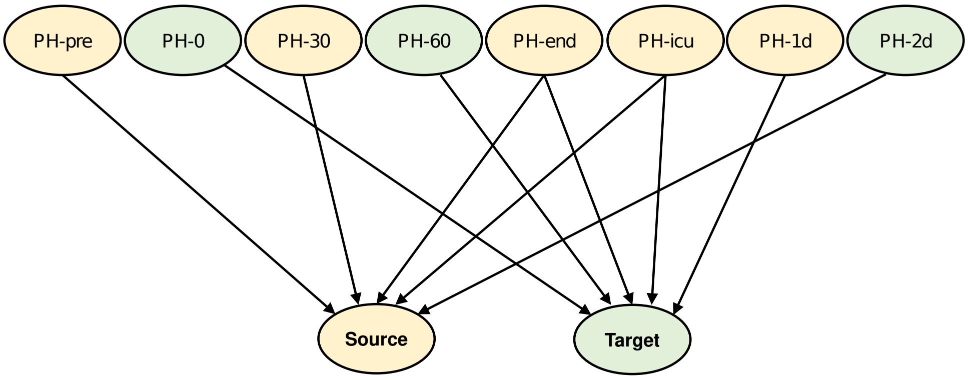


Fig. 2: PH feature division.

dominant eigenvalues of $(\widetilde{K}L + \lambda I)^{-1}\widetilde{K}H\widetilde{K}$. Correspondingly, $W^{\top}\widetilde{K}$ is the embedding matrix of samples into the latent space, with the j-th column $\left[W^{\top}\widetilde{K}\right]_j$ showing the latent vector of x_j.

TCA maps both domains to the same Reproducing Kernel Hilbert Space (RKHS) and adapts the marginal distributions in this RKHS. TCA assumes that when the marginal distribution adaptation is achieved, the two domains also satisfy the conditional probability distribution adaptation [17].

Given the small size but high dimensionality of the available liver transplantation dataset, we employ transfer learning to augment the size of our training set. The features of liver transplantation samples exhibit significant temporal characteristics and span various time intervals. Therefore, we extract corresponding medical observations along important time periods such as before, during, and after surgery, and then partition the dataset into the source and target domains. The liver transplant dataset has 54 common features and 394 time series features. Common features are obtained by a single measurement, such as age, height, and operation time (minutes), etc. Time series features are medical indicators that are measured multiple times, such as the PH value from blood gas analysis, which are measured several times during the perioperative period.

Fig. 2 illustrates an example of feature division. PH is a time series feature. PH-pre, PH-0, PH-30, PH-60, PH-end, PH-ICU, PH-1d, and PH-2d respectively refer to the blood gas PH values measured before the surgery, at the beginning of the surgery, 30 minutes after the surgery, 60 minutes after the surgery, at the end of the surgery, at the time of entering the ICU, one day after the surgery, and two days after the surgery. As shown in Fig. 2, we separate the PH values at adjacent moments and place them in the source and target domains, respectively. There are two moments that are particularly important to the operation, so we place the corresponding PH values (PH-end and PH-ICU) in the two domains. Additionally, the common features such as the Age and BMI are also placed in both domains in this study.

The first module in the proposed framework is the feature-based transfer learning module, illustrated in Fig. 3. The training set $\left(X^{train}\right)$ comprises the entire source domain data (S) and 2/3 of the target domain data (T), while the testing set (X^{test}) comprises the remaining 1/3 of the target domain data. Since this module uses unsupervised TCA, it does not require the labels of the training or testing sets. TCA projects the original features of the source and target domains into a shared latent space, which helps to reduce the domain shift between the two domains. The resulting latent features are then utilized as input to the downstream prediction model. In our case, the matrix $\widetilde{K}$ in formula (6) is generated from the training set, and it is formulated as $\widetilde{K} = \begin{bmatrix} K_{S,S} & ,K_{S,\frac{2}{3}T} \\ K_{\frac{2}{3}T,S} & ,K_{\frac{2}{3}T,\frac{2}{3}T} \end{bmatrix}$. L is defined as:

$$L_{ij} = \begin{cases} \frac{1}{n_S^2} & x_i, x_j \in X_S \\ \frac{1}{n_{\frac{2}{3}}^2} & x_i, x_j \in X_{\frac{2}{3}T} \\ -\frac{1}{n_S n_{\frac{2}{3}T}} & \text{otherwise} \end{cases} \tag{8}$$

As discussed afore, we can get the parametric matrix $W \in \mathbb{R}^{n \times m}$ with the TCA component and the training set, where n corresponds to the column dimension of the matrix $\widetilde{K}$, and $m \ll n$. With matrix W, we can also obtain the liver transplantation cases embedding in the latent space, which is $W^T \widetilde{K}$. The embedding vector of x_j equals to the j th column $\left[W^T \widetilde{K}\right]_j$. To prevent data leakage, we transfer the testing data into the same latent space with matrix W. First, we calculate the distance between the training and testing sets using a Radial Basis Function (RBF) kernel function. The resulting distance matrix is denoted as $K \in \mathbb{R}^{(n_1+n_2) \times n_3}$, where n_1 is the size of the source domain, n_2 is the size of two-thirds of the target domain, and n_3 equals to the size of one-third of the target domain. So, $n = n_1 + n_2$, which is equal to the size of the training set. n_3 is the size of the testing set. x_j^{train} and x_i^{test} represent data samples from the training set and testing set, respectively.

$$K^\top = \begin{bmatrix} \kappa\left(\boldsymbol{x}_1^{train}, \boldsymbol{x}_1^{test}\right) & \cdots & \kappa\left(\boldsymbol{x}_{n_1+n_2}^{train}, \boldsymbol{x}_1^{test}\right) \\ \vdots & & \vdots \\ \kappa\left(\boldsymbol{x}_1^{train}, \boldsymbol{x}_i^{test}\right) & \cdots & \kappa\left(\boldsymbol{x}_{n_1+n_2}^{train}, \boldsymbol{x}_i^{test}\right) \\ \vdots & & \vdots \\ \kappa\left(\boldsymbol{x}_1^{train}, \boldsymbol{x}_{n_3}^{test}\right) & \cdots & \kappa\left(\boldsymbol{x}_{n_1+n_2}^{train}, \boldsymbol{x}_{n_3}^{test}\right) \end{bmatrix} \tag{9}$$

Correspondingly, the embedding of testing data can be provided by $W^\top K$. The embedding of the ith data sample x_i^{test} in testing set is provided by column $\left[W^\top K\right]_i$. This aids the model in achieving better generalization when faced with new instances of postoperative complications.

In this module, we transfer the available dataset of liver transplantation to the same latent space to enlarge the training set. Simultaneously, dimensionality reduction is performed by introducing matrix $W \in \mathbb{R}^{n \times m}$ ($m \ll n$), which solves the problem of high-dimensional feature space of liver transplantation.

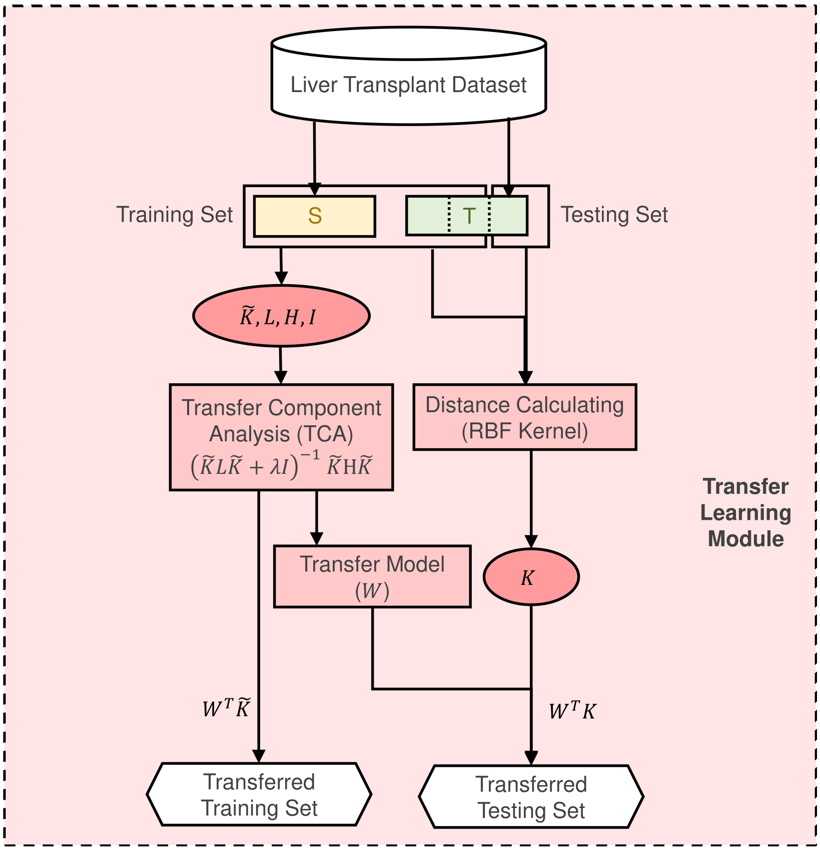


Fig. 3: Transfer learning process.

3.1.2 Heterogeneous Domain Adaptation (HDA)

When the feature sets of the domains have very little in common, heterogeneous transfer learning is used. In order to mitigate the differences and divergences in features and distributions between the domains, Li et al. [25] developed a progressive alignment approach.

By introducing the concept of a shared dictionary, a new transferable feature space can be obtained for two domains. In this new latent space, the features from different domains are linked together, allowing knowledge to be transferred from the original domains. This approach progressively reduces the distribution divergence between the two domains in the latent space, while maintaining local consistency. The key to reducing data distribution divergence is the use of a shared codebook **B**. Li et al. [25] formulated an optimization problem as shown in formula (10).

$$\min_{\mathbf{B},\mathbf{S}} \underbrace{\underbrace{\mathcal{C}_1\left(\mathbf{X_s},\mathbf{X_t},\mathbf{B},\mathbf{S}\right)}_{\text{feature alignment}} + \underbrace{\alpha\mathcal{C}_2\left(\mathbf{S_s},\mathbf{S_t}\right)}_{\text{distribution alignment}}}_{\text{progressive alignment}} + \underbrace{\beta\Omega(\mathbf{S})}_{\text{constraint}} \tag{10}$$

where $\alpha > 0$ and $\beta > 0$ are two hyperparameters. $\mathbf{X_s}$ and $\mathbf{X_t}$ represent the source and target domain data, respectively. $\mathbf{S_s}$ and $\mathbf{S_t}$ are new feature representations for $\mathbf{X_s}$ and $\mathbf{X_t}$, separately. $\mathbf{S}$ consists of both $\mathbf{S_s}$ and $\mathbf{S_t}$. The shared codebook $\mathbf{B}$ is used to acquire a transferable latent space to reduce the data distribution divergence. The optimization function includes three parts: the feature alignment $\mathcal{C}_1$, the distribution alignment $\mathcal{C}_2$, and the constraint (Ω). $\mathcal{C}_1$ learns the new transferable features $\mathbf{S}$, $\mathcal{C}_2$ aligns distributions based on new features $\mathbf{S}$, and Ω is used for regularization.

In HDA, the same features used in TCA for liver transplantation are used. The transferred features for the training set and testing set are obtained separately. The experimental section undertakes a comparative analysis of HDA, TCA, LASSO, and Principal Component Analysis (PCA). HDA is more appropriate than TCA for situations where the feature spaces of the domains are dissimilar. Since this study focuses on liver transplant data where the two domains exhibit similarity in feature and the label space, we primarily use TCA to construct the complication prediction model.

3.2 The Prediction of Complications

Given the limited number of liver transplant samples, it is easy to get overfitting with deep learning. Therefore, it is more suitable to use traditional machine learning methods to build classification models.

After reducing the dimensionality of the original dataset using transfer learning, SVM is used on the training set, as depicted in Fig. 3. SVM is a conventional machine learning method grounded in the theory of statistical learning, adhering to the principle of structural risk minimization [26]. It is robust and less likely to get stuck in a local minimum even with limited training data, and it is commonly used in similar biomedical applications [27].

We trained a single model for each kind of complication using SVM. The inputs for the models include the transferred training set obtained from TCA and the ground-truth labels of the training samples. The model specific to each complication performs binary classification on a transferable feature space, and the output is a prediction of whether the complication will occur or not. Fig. 4 illustrates the training and testing process of the proposed model. The complication i, $i \in \{1, 2, \ldots, 9\}$, belongs to a set of common postoperative complications of liver transplantation, including Complication Pleural Effusion, High INR, High ALT or AST, High TBIL, Pneumonia, Postoperative Bleeding, Postoperative Infection, Biliary Complications, and PNF.

3.3 Feature Impact Analysis

It is important for physicians to have clear explanations for models when making decisions about postoperative complications. In this work, we introduce the SHAP method to explain the proposed prediction model for complications by calculating feature importance. SHAP (SHapley Additive exPlanations) is a game theory-based

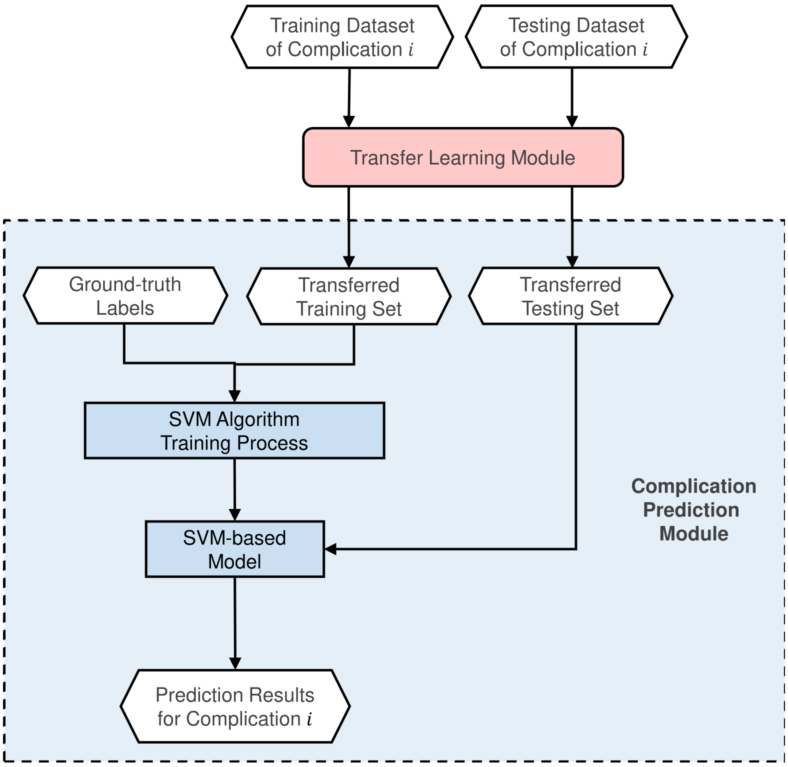

Training Dataset of Complication i
Testing Dataset of Complication i
Transfer Learning Module
Ground-truth Labels
Transferred Training Set
Transferred Testing Set
SVM Algorithm Training Process
SVM-based Model
Complication Prediction Module
Prediction Results for Complication i

Fig. 4: The machine learning-based models for complication prediction.

theory that is used to provide an explanation for the output of machine learning models. It uses Shapley values to link optimal credit allocation to local explanations. There are many forms of SHAP, but the Kernel SHAP is the most universal and can be utilized with any machine learning model. This is why we choose to use Kernel SHAP as the first form of SHAP in our work.

Kernel SHAP employs a conjunction of Linear LIME [28] and Shapley Values. The Shapley value explanation is represented using a method of an additive feature attribution, namely, a linear model. To find explanation $\xi(x)$ for each sample x, Kernel SHAP minimizes the following objective function as stated in formula (11). It defines the explanation $\xi(x)$ to be a model $g \in G$, in which G is a group of linearly interpretable models. $f : \mathbb{R}^M \to \mathbb{R}$ is the model being explained, and Ω measures the model complexity. M represents the number of input variables. The formula (12) defines the locally weighted square loss function $\mathcal{L}$. $x \in \mathbb{R}^M$ is the original representation of a sample being explained, and $x' \in \{0,1\}^M$ denotes a binary vector for its simple representation. $z' \in \{0,1\}^M$ is a perturbed sample, which includes some non-zero elements of x'. $z \in \mathbb{R}^M$ is the corresponding instance expressed by the original variables. Define $h_x(z')$ to be a function that maps z' back to the original representation z with the help of the original input x. The explanation model $g(z')$ is defined as a linear function of binary variables stated in formula (13), where $\phi_i \in \mathbb{R}$. $\pi_{x'}(z')$, defined by formula (14), represents the weight of the perturbed sample z', in which $|z'|$ is the number of nonzero entries of z'. A detailed description of Kernel SHAP is provided by paper [10]. From the obtained solution of $\xi(x)$, we can get a set $\{\phi_j \mid j = 1, 2, \ldots, M\}$, which indicates the importance of the corresponding feature j of sample x. For the entire set of original input dataset $\{x_i \in X \mid i = 1, 2, \ldots, n\}$, the global feature importance (I_j) of feature j is the average value of ϕ_j, given by Equation (15).

$$\xi(x) = \underset{g \in G}{\operatorname{argmin}} \mathcal{L}\left(f, g, \pi_x\right) + \Omega(g) \tag{11}$$

$$L\left(f, g, \pi_{x'}\right) = \sum_{z' \in Z} \left[f\left(h_x\left(z'\right)\right) - g\left(z'\right)\right]^2 \pi_{x'}\left(z'\right) \tag{12}$$

$$g\left(z'\right) = \phi_0 + \sum_{i=1}^{M} \phi_i z'_i \tag{13}$$

$$\pi_{x'}\left(z'\right) = \frac{M-1}{C_M^{|z'|} |z'| \left(M - |z'|\right)} \tag{14}$$

$$I_j = \frac{1}{n} \sum_{i=1}^{n} |\phi_j^{(i)}| \tag{15}$$

In this work, we use Kernel SHAP to provide a medical interpretation of the proposed model by calculating the feature importance of postoperative complications of liver transplantation. As mentioned earlier, the original input features are transformed into a latent feature space by TCA, and it is not possible to directly explain the original features. Therefore, we propose a model wrapper that combines the transfer learning module and the complication prediction model. As shown in Fig. 5, the wrapper eliminates the effect of feature transformation on the interpreter, and the output of the interpreter is a set of importance scores that explain the original features through the latent features. This allows us to directly interpret the original features of the input data.

4 Experimental Results and Analysis

This section presents the experimental evaluation of the proposed method for predicting postoperative complications of liver transplantation. Subsection 4.1 provides an elaborate overview of the dataset and describes the data preprocessing techniques employed. The experimental setup is outlined in Subsection 4.2. Subsection 4.3 presents the performance analysis of the proposed method in predicting complications, while Subsection 4.4 offers a comprehensive feature impact analysis. Additionally, Subsection 4.5 presents ablation studies conducted on the proposed method to verify its accuracy and effectiveness.

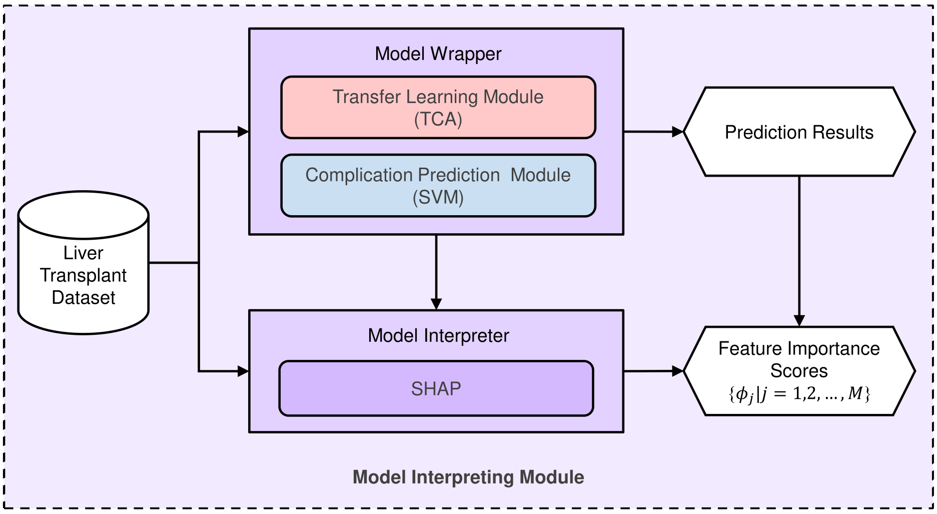

Model Wrapper
Transfer Learning Module (TCA)
Complication Prediction Module (SVM)
Prediction Results
Liver Transplant Dataset
Model Interpreter
SHAP
Feature Importance Scores $\{\phi_j | j = 1,2,\ldots,M\}$
Model Interpreting Module

Fig. 5: The framework of the interpreter.

4.1 Liver Transplant Dataset

The dataset employed in this study comprises perioperative medical records of liver transplant patients treated at Beijing Chaoyang Hospital between 2016 and 2019. To ensure privacy, the data has been de-identified. The collection of data involves three electronic systems: the anesthesia order monitoring system, the medical record system, and the ICU nursing record system.

The dataset comprises medical information of 425 liver transplant patients who experienced confirmed postoperative complications. Table 1 displays the demographic characteristics of 425 patients, including their median age of 53 years (with a standard deviation, denoted as s.d., of 10.5). Among these patients, 78.6% are male, the average body mass index (BMI) is 24.5 (with a standard deviation of 3.6), and 60.8% are hepatitis B carriers.

Table 1: Demographic information of subjects in dataset

Age (median(s.d.))	Female (%)	BMI (mean(s.d.))	Hepatitis B carrier (%)
53 (10.5)	21.4%	24.5(3.6)	60.8%

Each instance in the dataset comprises 448 features, encompassing various categories such as General data, Blood routine tests, Biochemical tests, Blood gas analysis, Blood coagulation, Intraoperative data, Postoperative transfusion records, and Postoperative outcomes before hospital discharge. The label value ”1” signifies the occurrence of a postoperative complication, while the label value ”0” indicates the absence of a corresponding complication. For a comprehensive illustration, a detailed example is provided in Appendix A.

This paper discusses nine complications, which are outlined in Table 2: Pleural effusion, High INR, High ALT or AST, High TBIL, Pneumonia, Postoperative bleeding, Postoperative infection, Biliary complications, and PNF. As depicted in Table 2, there is an imbalance in the proportion of positive and negative samples, particularly for certain complications, such as Pneumonia, Postoperative bleeding, Postoperative infection, Biliary complications, and PNF, where the proportion of positive instances is below 22%. The liver transplant dataset exhibits characteristics of being imbalanced and high-dimensional, further compounded by the limited number of samples available. These dataset characteristics can exert a detrimental impact on the model training process. Consequently, we employ transfer learning as a means to data augmentation.

Table 2: Description of liver transplant dataset

Complications	Size of Positive Samples	Proportion of Positive Samples
Pleural Effusion	319	75%
High INR	246	42%
High ALT or AST	117	28%
High TBIL	94	22%
Pneumonia	89	21%
Postoperative Bleeding	22	5%
Postoperative Infection	21	5%
Biliary Complications	10	2%
PNF	9	2%

Alongside the challenges posed by imbalance and high dimensionality, the dataset also exhibits missing values for certain features. Only features with a missing rate below 60% were retained for further analysis. To address the missing values, we employed random forests to predict and impute values for these features. Moreover, to mitigate the influence of varying scales among features during model training, we normalized the feature values.

4.2 Setup

To construct our training set, we utilized all available data from the source domain, supplemented by 2/3 of the data from the target domain. The remaining 1/3 of the target domain data was reserved for our testing set. In order to ensure reliable estimation, we assessed the performance of our classifier through 3-fold cross-validation.

To mitigate the influence of class imbalance during model training, we established a threshold of 22% and employed the Random Over Sampling (ROS) technique for oversampling if the imbalance exceeds this threshold. Additionally, we enhanced performance by reducing the dimensionality of the feature space from 456 to 50.

In our experiments, we compared the effectiveness of various data processing methods, including transfer learning methods such as TCA and HDA, the traditional dimensionality reduction technique PCA, and the feature selection method LASSO.

For the prediction of complications, we utilized Support Vector Machines (SVM) with Gaussian kernels, K-Nearest Neighbors (KNN) with 15 nearest neighbors, and eXtreme Gradient Boosting (XGBoost) with 60 decision trees. Additionally, we tuned the SVM optimal values of C and γ using grid search, where $C = 100$ and $\gamma = 100$ yielded the best accuracy. LASSO selected 10 features with the largest coefficients from 448 features for logistic regression. During the missing value-filling process, we used the random forest model with 600 decision trees (n-estimators), and at least one sample was required for an internal node split.

The evaluation metrics used in this work include precision (P), recall (R), and F-score (F_1), which are represented by Equations (16)-(18).

$$P = \frac{TP}{TP + FP} \tag{16}$$

$$R = \frac{TP}{TP + FN} \tag{17}$$

$$F_1 = \frac{2}{P^{-1} + R^{-1}} = \frac{2TP}{2TP + FN + FP} \tag{18}$$

where TP, TN, FP, and FN represent the count of true positives, true negatives, false positives, and false negatives, respectively.

4.3 Performance of Complication Prediction

We evaluated the performance of our proposed complication prediction method, which combines TCA and SVM, using 3-fold cross-validation as described in Section 3. Table 3 presents the statistical results for the prediction of nine liver transplant complications in terms of precision, recall, and F1 score. We implemented oversampling for the last five complications due to the low proportion of positive samples (less than 22%) in these cases. Especially for biliary complications, the number of positive samples is small. Therefore, the performance was superior when the prediction was successful for several positive samples. This raises concerns about the model stability when faced with new cases, as the limited number of available samples may not be representative of the wider population. Therefore, there is a need to measure the model robustness on new samples.

To evaluate the stability of the model, we calculated confidence intervals for prediction accuracy at a 95% confidence level, as presented in Table 4 and Fig. 6. The results indicate that the models have the potential to yield promising outcomes when applied to unknown samples. Notably, the prediction model for biliary complications demonstrated excellent performance across all samples, with both the upper and lower bounds of the confidence interval reaching 100%. This can be attributed primarily to the limited number of positive samples and the small number of specific features that exhibit higher contribution values, as depicted in Fig. 7(h). These findings suggest that the model effectively utilizes crucial features, such as postoperative red blood cell infusion volume and ascites volume, leading to comparable accuracy in predicting this complication.

Table 3: Performance of complication prediction model based on SVM and TCA

Complications	F1(Mean)	Precision(Mean)	Recall(Mean)
Pleural effusion	91.65%	86.85%	97.18%
High INR	83.73%	84.57%	82.93%
High ALT or AST	95.00%	92.75%	97.44%
High TBIL	90.37%	94.35%	87.13%
Pneumonia	80.26%	76.17%	85.44%
Postoperative bleeding	91.07%	91.07%	91.07%
Postoperative infection	95.24%	95.24%	95.24%
Biliary complications	100.00%	100.00%	100.00%
PNF	93.33%	100.00%	88.89%

Table 4: Confidence intervals of complication prediction models at a 95% confidence level

Complications	Accuracy	CI Lower	CI Upper
Pleural effusion	86.12%	80.23%	91.29%
High INR	79.05%	72.23%	85.64%
High ALT or AST	97.18%	94.36%	99.30%
High TBIL	94.83%	90.83%	97.89%
Pneumonia	89.64%	84.46%	94.35%
Postoperative bleeding	99.29%	98.12%	100.00%
Postoperative infection	98.82%	97.41%	99.77%
Biliary complications	100.00%	100.00%	100.00%
PNF	99.77%	99.30%	100.00%

4.4 Feature Impact

We used Shapley value analysis to evaluate the influence of individual inputs on the output probability of our model for each patient in the testing set. The Shapley values were calculated to provide an average impact on the overall model output, offering a more comprehensive understanding than an analysis at the level of individual predictions.

Figures 7(a)-7(i) depict the impact of features on each of the nine complication predictions. The figures show the top ten most important features for each prediction. As shown, the most significant common features are related to intraoperative conditions (such as the amount of input and total urine output), bio-chemical indicators (such as AST-post and ALT-post), and postoperative blood transfusion (such as postoperative plasma (total) and postoperative red blood cells (total)).

Some of the features have been proven by medical reports to be independent predictors for the occurrence of certain complications. For example, increased perioperative infusion of red blood cells has been reported to be related to more postoperative pulmonary complications, including pleural effusion, which is consistent with the output of our model [29–31]. Postoperative plasma (total) can provide proteins that may help to keep fluid in the vessels, thus reducing the production of hydrothorax [32]. For the

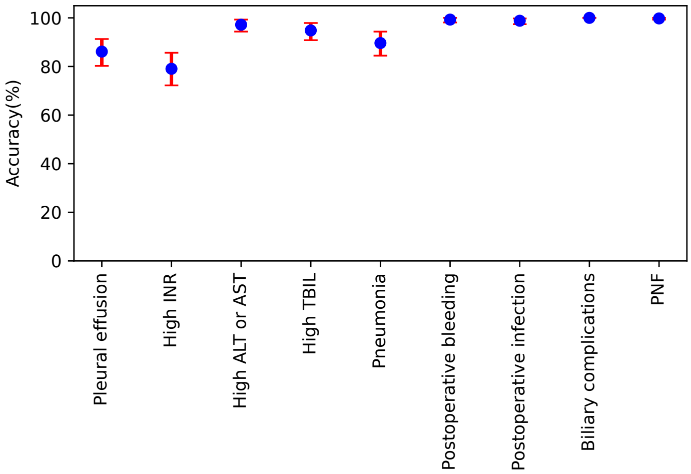


Fig. 6: The error bar plot with confidence intervals for the accuracy of complication prediction model based on SVM and TCA.

amount of input, it was found that a negative balance of fluid infusion is related to postoperative pulmonary complications [29, 33]. Some features are still under research or have not been noticed yet by doctors, although they may be significantly associated with these complications. Levels of glutamic oxaloacetic transaminase (AST) or glutamic pyruvic transaminase (ALT) were not reported to be predictors for hydrothorax. Actually, no related report was found. Since both are among the top ten features, we sent them for further evaluation by medical statistics. Since the number of positive samples of pleural effusion is more than other complications, more meaningful results can be obtained. We carried out medical statistics on the top 10 features of pleural effusion. The statistical results of multiple stepwise regression analysis are presented in Table 5. In the table, beta represents the standardized regression coefficient, SE represents the standard error of the coefficient, P represents the p-value of the coefficient, OR represents the odds ratio. In addition, we report the lower confidence limit (LowerCL) and upper confidence limit (UpperCL). Variables with a p-value less than 0.05 are considered statistically significant.

All five significant features selected by multiple stepwise regression analysis, as shown in Table 5, are included in the results obtained from our approach. Preliminary results shows that the overall fluid input during LT (Liver Transplantation), the volume of plasma infusion after LT, AST level on the second day after LT, ALT level right after surgery, and the volume of red cell infused after LT are independent predictors for the occurrence of pleural effusion after LT. However, the volume of ascites, urine, and 4% albumin infusion during LT, or the volume of postoperative red cell infusion is not significantly associated with pleural effusion in the statistical results.

Table 5: Multiple stepwise regression results for Pleural Effusion using the top features

Features	Beta	SE	P	OR	LowerCL	UpperCL
Amount of input	-0.5850	0.1629	0.0003	0.557	0.405	0.767
AST-1,2	-1.7301	0.6297	0.01	0.177	0.052	0.609
Postoperative plasma (total)	-0.3002	0.1233	0.0149	0.741	0.582	0.943
AST-post	0.6781	0.2018	0.0008	1.970	1.327	2.926
Red blood cells	0.3858	0.1709	0.0240	1.471	1.052	2.056

Nonetheless, they are still important as they participate in the perioperative fluid balance and indirectly affect the production of pleural effusion [31, 34, 35]. Perhaps they need some features as conditions or background to become significant. The important features found out by SHAP in our study can help verify previous medical reports and provide fresh objectives for future studies.

4.5 Ablation Study

The aim of this subsection is to evaluate the performance of the suggested architecture using a real dataset for liver transplantation. We begin by conducting an ablation study of the transfer learning modules. Next, we incorporate various machine-learning techniques into our framework and compare their results.

4.5.1 Ablation Study on Transfer Learning Modules

Tables 6 to 8 present the results of the suggested prediction model based on SVM, combined with TCA, HDA, and PCA, respectively. Furthermore, the tables also include the results of the prediction model based on Logistic Regression, incorporating LASSO. From the tables, it is evident that the prediction models utilizing TCA demonstrate the highest performance in terms of F1 score, precision, and recall. Notably, when the proportion of positive samples is below 22%, the performance of other comparison methods experiences a significant decrease. In contrast, the proposed method, which utilizes a homogeneous transfer learning algorithm such as TCA, proves to be efficient in predicting liver complications, especially in imbalanced scenarios.

Table 6: F1 comparison of models using TCA, HAD, PCA and LASSO

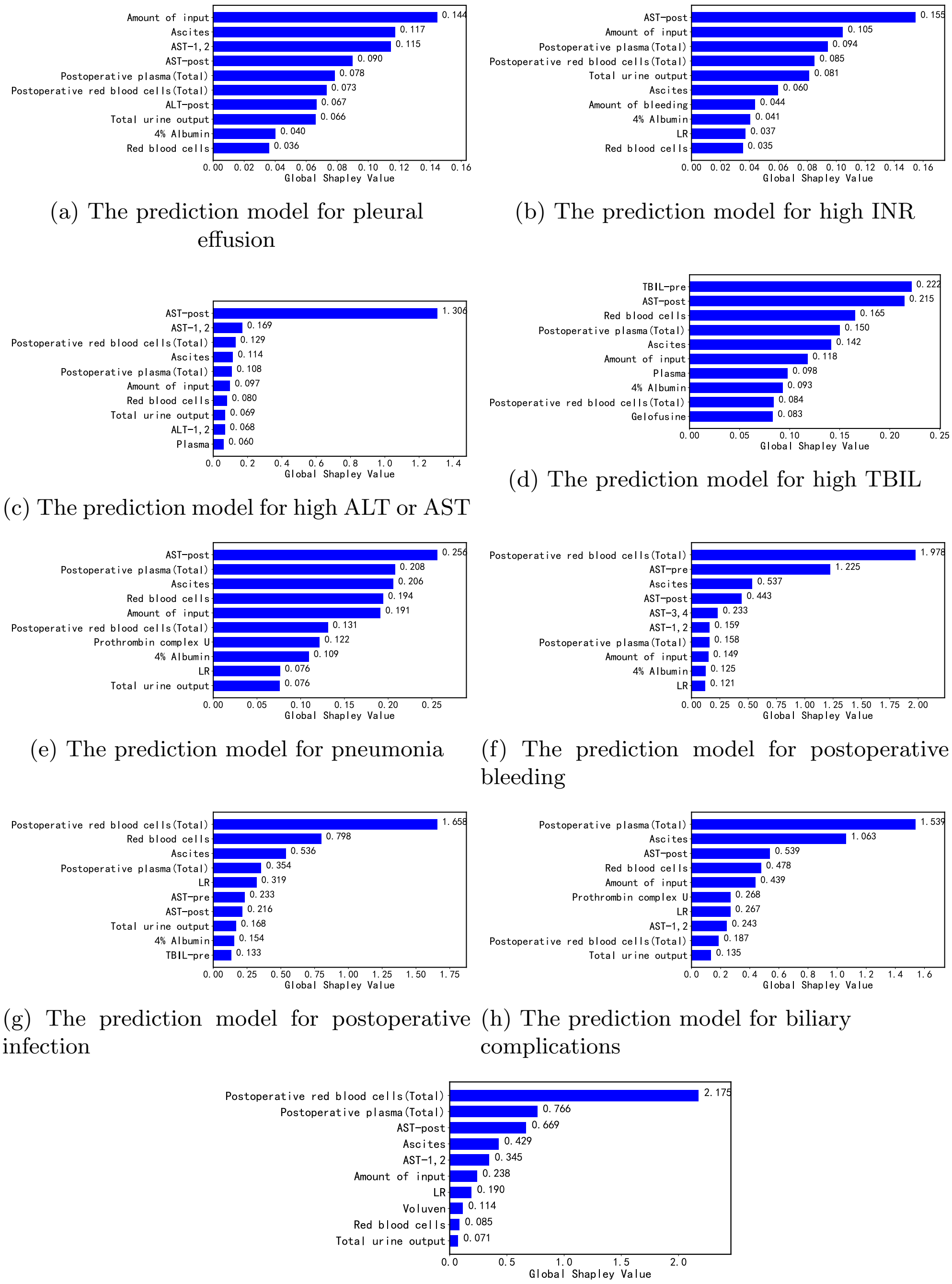
Complications	TCA+SVM	HDA+SVM	PCA+SVM	LASSO+LR
Pleural effusion	**91.65%**	85.75%	85.75%	80.73%
High INR	**83.73%**	73.32%	73.32%	74.44%
High ALT or AST	**95.00%**	0.00%	0.00%	68.28%
High TBIL	**90.37%**	0.00%	0.00%	63.54%
Pneumonia	**80.26%**	0.00%	0.00%	38.26%
Postoperative bleeding	**91.07%**	0.00%	0.00%	24.54%
Postoperative infection	**95.24%**	0.00%	0.00%	19.58%
Biliary complications	**100.00%**	0.00%	0.00%	9.52%
PNF	**93.33%**	0.00%	0.00%	9.52%

Table 7: Precision comparison of models using TCA, HAD, PCA and LASSO

Complications	TCA+SVM	HDA+SVM	PCA+SVM	LASSO+LR
Pleural effusion	**86.85%**	75.06%	75.06%	81.51%
High INR	**84.57%**	57.88%	57.88%	72.76%
High ALT or AST	**92.75%**	0.00%	0.00%	64.96%
High TBIL	**94.35%**	0.00%	0.00%	62.80%
Pneumonia	**76.17%**	0.00%	0.00%	38.24%
Postoperative bleeding	**91.07%**	0.00%	0.00%	27.38%
Postoperative infection	**95.24%**	0.00%	0.00%	19.05%
Biliary complications	**100.00%**	0.00%	0.00%	11.11%
PNF	**100.00%**	0.00%	0.00%	11.11%

Table 8: Recall comparison of models using TCA, HAD, PCA and LASSO

Complications	TCA+SVM	HDA+SVM	PCA+SVM	LASSO+LR
Pleural effusion	**97.18%**	100.00%	100.00%	80.05%
High INR	**82.93%**	100.00%	100.00%	76.68%
High ALT or AST	**97.44%**	0.00%	0.00%	73.91%
High TBIL	**87.13%**	0.00%	0.00%	64.39%
Pneumonia	**85.44%**	0.00%	0.00%	39.33%
Postoperative bleeding	**91.07%**	0.00%	0.00%	22.50%
Postoperative infection	**95.24%**	0.00%	0.00%	21.13%
Biliary complications	**100.00%**	0.00%	0.00%	8.33%
PNF	**88.89%**	0.00%	0.00%	8.33%



(a) The prediction model for pleural effusion

(b) The prediction model for high INR

(c) The prediction model for high ALT or AST

(d) The prediction model for high TBIL

(e) The prediction model for pneumonia

(f) The prediction model for postoperative bleeding

(g) The prediction model for postoperative infection

(h) The prediction model for biliary complications

(i) The prediction model for PNF

Fig. 7: The obtained top ten most important features for each complication prediction using TCA and SVM.

4.5.2 Ablation Study on Machine Learning Modules

Tables 9 through 11 provide a comparison of various machine learning methods when combined with Transfer Component Analysis (TCA). The results highlight the performance of these methods in terms of recall, precision, and F1 score. Overall, the findings indicate that SVM outperforms all other machine learning algorithms examined in this study, specifically on the real-world imbalanced and HDLSS (High Dimension and Low Sample Size) dataset. The proposed architecture exhibits exceptional performance compared to the alternatives, leading to the selection of SVM as the chosen classification method in this paper.

Table 9: F1 measure comparison of different prediction models combined with TCA

Methods	Pleural effusion	High INR	High ALT or AST	High TBIL	Pneumonia	Postoperative bleeding	Postoperative infection	Biliary complications	PNF
SVM	**91.65%**	**83.73%**	**95.00%**	**90.37%**	**80.26%**	**91.07%**	**95.24%**	**100.00%**	**93.33%**
XGB	89.73%	87.40%	88.88%	78.10%	70.89%	46.55%	39.83%	65.56%	16.67%
KNN	84.56%	71.97%	79.40%	40.02%	45.47%	37.98%	38.30%	28.79%	35.83%

Table 10: Precision comparison of different prediction models combined with TCA

Methods	Pleural effusion	High INR	High ALT or AST	High TBIL	Pneumonia	Postoperative bleeding	Postoperative infection	Biliary complications	PNF
SVM	**86.85%**	**84.57%**	**92.75%**	**94.35%**	76.17%	**91.07%**	**95.24%**	**100.00%**	**100.00%**
XGB	83.35%	82.41%	88.88%	88.61%	**81.49%**	57.94%	52.22%	**100.00%**	33.33%
KNN	76.00%	67.45%	83.25%	61.30%	34.90%	23.49%	23.71%	16.93%	21.91%

Table 11: Recall comparison of different prediction models combined with TCA

Methods	Pleural effusion	High INR	High ALT or AST	High TBIL	Pneumonia	Postoperative bleeding	Postoperative infection	Biliary complications	PNF
SVM	**97.18%**	82.93%	**97.44%**	**87.13%**	**85.44%**	91.07%	95.24%	**100.00%**	88.89%
XGB	**97.18%**	**93.09%**	88.89%	70.19%	62.91%	41.67%	33.33%	50.00%	11.11%
KNN	95.30%	77.24%	76.07%	29.77%	65.29%	**100.00%**	**100.00%**	**100.00%**	**100.00%**

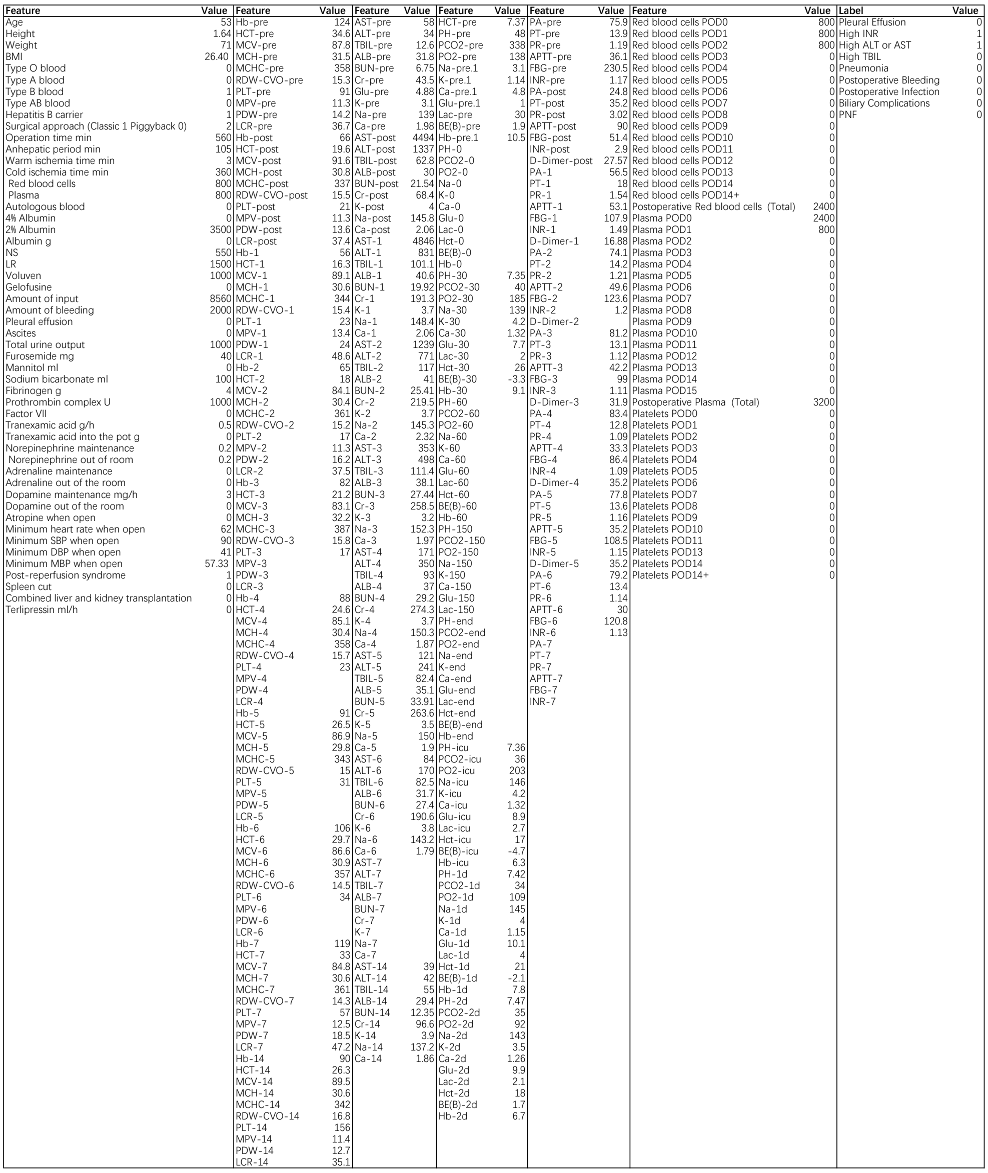
5 Conclusion

This paper presents a model for predicting postoperative complications of liver transplantation using transfer learning and machine learning techniques. The experiments demonstrate that the Support Vector Machine is well-suited for obtaining accurate predictions using a dataset of liver transplantation with a limited number of positive samples. Moreover, the implemented feature-based transfer learning module effectively reduces dimensionality and enables data augmentation, which mitigated the effects of class imbalance and small dataset size on model training. As a result, it significantly improves prediction performance concerning recall, precision, and F1 score.

Furthermore, this paper adopts the SHAP (Shapley Additive exPlanations) method for model interpretation. To ensure the interpretation is not affected by feature transformation, a model wrapper is proposed that combines the transfer learning module and the complication prediction module. Experimental results demonstrate that the proposed method can directly interpret the original features of the input data.

The identified key features have the potential to assist doctors in making accurate diagnoses and can provide new directions for future medical research. However, the validity of model interpretation for certain complications is limited due to the lack of positive samples. To address this, a larger number of liver transplant cases are required to enrich the dataset. While the proposed model complements the approach based on medical reports, it is essential to validate the identified features through clinical practices and experience. Currently, the key features have been validated for complications with a high number of positive samples, and the validation of complications with a low occurrence rate will be conducted in future studies.

Appendix A Instance Example



Feature	Value	Feature	Value	Feature	Value	Feature	Value	Feature	Value	Feature	Value	Label	Value
Age	53	Hb-pre	124	AST-pre	58	HCT-pre	7.37	PA-pre	75.9	Red blood cells POD0	800	Pleural Effusion	0
Height	1.64	HCT-pre	34.6	ALT-pre	34	PH-pre	48	PT-pre	13.9	Red blood cells POD1	800	High INR	1
Weight	71	MCV-pre	87.8	TBIL-pre	12.6	PCO2-pre	338	PR-pre	1.19	Red blood cells POD2	800	High ALT or AST	1
BMI	26.40	MCH-pre	31.5	ALB-pre	31.8	PO2-pre	138	APTT-pre	36.1	Red blood cells POD3	0	High TBIL	0
Type O blood	0	MCHC-pre	358	BUN-pre	6.75	Na-pre.1	3.1	FBG-pre	230.5	Red blood cells POD4	0	Pneumonia	0
Type A blood	0	RDW-CVO-pre	15.3	Cr-pre	43.5	K-pre.1	1.14	INR-pre	1.17	Red blood cells POD5	0	Postoperative Bleeding	0
Type B blood	1	PLT-pre	91	Glu-pre	4.88	Ca-pre.1	4.8	PA-post	24.8	Red blood cells POD6	0	Postoperative Infection	0
Type AB blood	0	MPV-pre	11.3	K-pre	3.1	Glu-pre.1	1	PT-post	35.2	Red blood cells POD7	0	Biliary Complications	0
Hepatitis B carrier	1	PDW-pre	14.2	Na-pre	139	Lac-pre	30	PR-post	3.02	Red blood cells POD8	0	PNF	0
Surgical approach (Classic 1 Piggyback 0)	2	LCR-pre	36.7	Ca-pre	1.98	BE(B)-pre	1.9	APTT-post	90	Red blood cells POD9	0		
Operation time min	560	Hb-post	66	AST-post	4494	Hb-pre.1	10.5	FBG-post	51.4	Red blood cells POD10	0		
Anhepatic period min	105	HCT-post	19.6	ALT-post	1337	PH-0		INR-post	2.9	Red blood cells POD11	0		
Warm ischemia time min	3	MCV-post	91.6	TBIL-post	62.8	PCO2-0		D-Dimer-post	27.57	Red blood cells POD12	0		
Cold ischemia time min	360	MCH-post	30.8	ALB-post	30	PO2-0		PA-1	56.5	Red blood cells POD13	0		
Red blood cells	800	MCHC-post	337	BUN-post	21.54	Na-0		PT-1	18	Red blood cells POD14	0		
Plasma	800	RDW-CVO-post	15.5	Cr-post	68.4	K-0		PR-1	1.54	Red blood cells POD14+	0		
Autologous blood	0	PLT-post	21	K-post	4	Ca-0		APTT-1	53.1	Postoperative Red blood cells (Total)	2400		
4% Albumin	0	MPV-post	11.3	Na-post	145.8	Glu-0		FBG-1	107.9	Plasma POD0	2400		
2% Albumin	3500	PDW-post	13.6	Ca-post	2.06	Lac-0		INR-1	1.49	Plasma POD1	800		
Albumin g	0	LCR-post	37.4	AST-1	4846	Hct-0		D-Dimer-1	16.88	Plasma POD2	0		
NS	550	Hb-1	56	ALT-1	831	BE(B)-0		PA-2	74.1	Plasma POD3	0		
LR	1500	HCT-1	16.3	TBIL-1	101.1	Hb-0		PT-2	14.2	Plasma POD4	0		
Voluven	1000	MCV-1	89.1	ALB-1	40.6	PH-30	7.35	PR-2	1.21	Plasma POD5	0		
Gelofusine	0	MCH-1	30.6	BUN-1	19.92	PCO2-30	40	APTT-2	49.6	Plasma POD6	0		
Amount of input	8560	MCHC-1	344	Cr-1	191.3	PO2-30	185	FBG-2	123.6	Plasma POD7	0		
Amount of bleeding	2000	RDW-CVO-1	15.4	K-1	3.7	Na-30	139	INR-2	1.2	Plasma POD8	0		
Pleural effusion	0	PLT-1	23	Na-1	148.4	K-30	4.2	D-Dimer-2		Plasma POD9	0		
Ascites	0	MPV-1	13.4	Ca-1	2.06	Ca-30	1.32	PA-3	81.2	Plasma POD10	0		
Total urine output	1000	PDW-1	24	AST-2	1239	Glu-30	7.7	PT-3	13.1	Plasma POD11	0		
Furosemide mg	40	LCR-1	48.6	ALT-2	771	Lac-30	2	PR-3	1.12	Plasma POD12	0		
Mannitol ml	0	Hb-2	65	TBIL-2	117	Hct-30	26	APTT-3	42.2	Plasma POD13	0		
Sodium bicarbonate ml	100	HCT-2	18	ALB-2	41	BE(B)-30	-3.3	FBG-3	99	Plasma POD14	0		
Fibrinogen g	4	MCV-2	84.1	BUN-2	25.41	Hb-30	9.1	INR-3	1.11	Plasma POD15	0		
Prothrombin complex U	1000	MCH-2	30.4	Cr-2	219.5	PH-60		D-Dimer-3	31.9	Postoperative Plasma (Total)	3200		
Factor VII	0	MCHC-2	361	K-2	3.7	PCO2-60		PA-4	83.4	Platelets POD0	0		
Tranexamic acid g/h	0.5	RDW-CVO-2	15.2	Na-2	145.3	PO2-60		PT-4	12.8	Platelets POD1	0		
Tranexamic acid into the pot g	0	PLT-2	17	Ca-2	2.32	Na-60		PR-4	1.09	Platelets POD2	0		
Norepinephrine maintenance	0.2	MPV-2	11.3	AST-3	353	K-60		APTT-4	33.3	Platelets POD3	0		
Norepinephrine out of room	0.2	PDW-2	16.2	ALT-3	498	Ca-60		FBG-4	86.4	Platelets POD4	0		
Adrenaline maintenance	0	LCR-2	37.5	TBIL-3	111.4	Glu-60		INR-4	1.09	Platelets POD5	0		
Adrenaline out of the room	0	Hb-3	82	ALB-3	38.1	Lac-60		D-Dimer-4	35.2	Platelets POD6	0		
Dopamine maintenance mg/h	3	HCT-3	21.2	BUN-3	27.44	Hct-60		PA-5	77.8	Platelets POD7	0		
Dopamine out of the room	0	MCV-3	83.1	Cr-3	258.5	BE(B)-60		PT-5	13.6	Platelets POD8	0		
Atropine when open	0	MCH-3	32.2	K-3	3.2	Hb-60		PR-5	1.16	Platelets POD9	0		
Minimum heart rate when open	62	MCHC-3	387	Na-3	152.3	PH-150		APTT-5	35.2	Platelets POD10	0		
Minimum SBP when open	90	RDW-CVO-3	15.8	Ca-3	1.97	PCO2-150		FBG-5	108.5	Platelets POD11	0		
Minimum DBP when open	41	PLT-3	17	AST-4	171	PO2-150		INR-5	1.15	Platelets POD13	0		
Minimum MBP when open	57.33	MPV-3		ALT-4	350	Na-150		D-Dimer-5	35.2	Platelets POD14	0		
Post-reperfusion syndrome	1	PDW-3		TBIL-4	93	K-150		PA-6	79.2	Platelets POD14+	0		
Spleen cut	0	LCR-3		ALB-4	37	Ca-150		PT-6	13.4				
Combined liver and kidney transplantation	0	Hb-4	88	BUN-4	29.2	Glu-150		PR-6	1.14				
Terlipressin ml/h	0	HCT-4	24.6	Cr-4	274.3	Lac-150		APTT-6	30				
		MCV-4	85.1	K-4	3.7	PH-end		FBG-6	120.8				
		MCH-4	30.4	Na-4	150.3	PCO2-end		INR-6	1.13				
		MCHC-4	358	Ca-4	1.87	PO2-end		PA-7					
		RDW-CVO-4	15.7	AST-5	121	Na-end		PT-7					
		PLT-4	23	ALT-5	241	K-end		PR-7					
		MPV-4		TBIL-5	82.4	Ca-end		APTT-7					
		PDW-4		ALB-5	35.1	Glu-end		FBG-7					
		LCR-4		BUN-5	33.91	Lac-end		INR-7					
		Hb-5	91	Cr-5	263.6	Hct-end							
		HCT-5	26.5	K-5	3.5	BE(B)-end							
		MCV-5	86.9	Na-5	150	Hb-end							
		MCH-5	29.8	Ca-5	1.9	PH-icu	7.36						
		MCHC-5	343	AST-6	84	PCO2-icu	36						
		RDW-CVO-5	15	ALT-6	170	PO2-icu	203						
		PLT-5	31	TBIL-6	82.5	Na-icu	146						
		MPV-5		ALB-6	31.7	K-icu	4.2						
		PDW-5		BUN-6	27.4	Ca-icu	1.32						
		LCR-5		Cr-6	190.6	Glu-icu	8.9						
		Hb-6	106	K-6	3.8	Lac-icu	2.7						
		HCT-6	29.7	Na-6	143.2	Hct-icu	17						
		MCV-6	86.6	Ca-6	1.79	BE(B)-icu	-4.7						
		MCH-6	30.9	AST-7		Hb-icu	6.3						
		MCHC-6	357	ALT-7		PH-1d	7.42						
		RDW-CVO-6	14.5	TBIL-7		PCO2-1d	34						
		PLT-6	34	ALB-7		PO2-1d	109						
		MPV-6		BUN-7		Na-1d	145						
		PDW-6		Cr-7		K-1d	4						
		LCR-6		K-7		Ca-1d	1.15						
		Hb-7	119	Na-7		Glu-1d	10.1						
		HCT-7	33	Ca-7		Lac-1d	4						
		MCV-7	84.8	AST-14	39	Hct-1d	21						
		MCH-7	30.6	ALT-14	42	BE(B)-1d	-2.1						
		MCHC-7	361	TBIL-14	55	Hb-1d	7.8						
		RDW-CVO-7	14.3	ALB-14	29.4	PH-2d	7.47						
		PLT-7	57	BUN-14	12.35	PCO2-2d	35						
		MPV-7	12.5	Cr-14	96.6	PO2-2d	92						
		PDW-7	18.5	K-14	3.9	Na-2d	143						
		LCR-7	47.2	Na-14	137.2	K-2d	3.5						
		Hb-14	90	Ca-14	1.86	Ca-2d	1.26						
		HCT-14	26.3			Glu-2d	9.9						
		MCV-14	89.5			Lac-2d	2.1						
		MCH-14	30.6			Hct-2d	18						
		MCHC-14	342			BE(B)-2d	1.7						
		RDW-CVO-14	16.8			Hb-2d	6.7						
		PLT-14	156										
		MPV-14	11.4										
		PDW-14	12.7										
		LCR-14	35.1										

References

[1] Deok-Bog Moon SGL (2009) Liver transplantation. Gut Liver 3(3):145–165. https://doi.org/10.5009/gnl.2009.3.3.145

[2] Bertacco A, Barbieri S, Guastalla G, et al (2019) Risk factors for early mortality in liver transplant patients. Transplant Proc 51(1):179–183. https://doi.org/10.

1016/j.transproceed.2018.06.025

[3] Salviano MEM, Lima AS, Tonelli IS, et al (2019) Primary liver graft dysfunction and non-function: integrative literature review. Rev Col Bras Cir 46. https://doi.org/10.1590/0100-6991e-20192039

[4] Uemura T, Randall HB, Sanchez EQ, et al (2007) Liver retransplantation for primary nonfunction: analysis of a 20-year single-center experience. Liver Transpl 13(2):227–233. https://doi.org/10.1002/lt.20992

[5] Sarhan MD, Osman AM, Mohamed MA, et al (2017) Biliary complications in recipients of living-donor liver transplant: a single-center review of 120 patients. Exp Clin Transplant 15(6):648–57. https://doi.org/10.6002/ect.2016.0210

[6] Craig EV, Heller MT (2021) Complications of liver transplant. Abdom Radiol 46(1):43–67. https://doi.org/10.1007/s00261-019-02340-5

[7] Liu CL, Soong RS, Lee WC, et al (2020) Predicting short-term survival after liver transplantation using machine learning. Sci Rep 10(1):1–10. https://doi.org/10.1038/s41598-020-62387-z

[8] Kantidakis G, Putter H, Lancia C, et al (2020) Survival prediction models since liver transplantation-comparisons between cox models and machine learning techniques. BMC Med Res Methodol 20(1):1–14. https://doi.org/10.1186/s12874-020-01153-1

[9] Ray S (2021) A survey on application of machine learning algorithms in cancer prediction and prognosis. In: Data Management, Analytics and Innovation: Proceedings of ICDMAI 2020, Volume 1, pp 349–361, https://doi.org/10.1007/978-981-15-5616-6_25

[10] Lundberg SM, Lee SI (2017) A unified approach to interpreting model predictions. Adv Neur In 30. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf

[11] Asadi H, Kok HK, Looby S, et al (2016) Outcomes and complications after endovascular treatment of brain arteriovenous malformations: a prognostication attempt using artificial intelligence. World Neurosurg 96:562–569. https://doi.org/10.1016/j.wneu.2016.09.086

[12] Hope TM, Seghier ML, Leff AP, et al (2013) Predicting outcome and recovery after stroke with lesions extracted from mri images. Neuroimage Clin 2:424–433. https://doi.org/10.1016/j.nicl.2013.03.005

[13] Li Y, Liu X, Jiang Y, et al (2021) Low preoperative prealbumin predicts the prevalence of complications following liver transplantation. BMC Gastroenterol 21(1):1–9. https://doi.org/10.1186/s12876-021-01818-1

[14] Al-Stouhi S, Reddy CK (2016) Transfer learning for class imbalance problems with inadequate data. Knowl Inf Syst 48(1):201–228. https://doi.org/10.1007/s10115-015-0870-3

[15] Pan SJ, Yang Q (2010) A survey on transfer learning. IEEE Trans Knowl Data Eng 22(10):1345–1359. https://doi.org/10.1109/TKDE.2009.191

[16] Liu T, Yang Q, Tao D (2017) Understanding how feature structure transfers in transfer learning. In: IJCAI, pp 2365–2371, https://doi.org/10.24963/ijcai.2017/329

[17] Pan SJ, Tsang IW, Kwok JT, et al (2010) Domain adaptation via transfer component analysis. IEEE Trans Neural Netw 22(2):199–210. https://doi.org/10.1109/tnn.2010.2091281

[18] Ghosn J, Bengio Y (2003) Bias learning, knowledge sharing. IEEE Trans Neural Netw 14(4):748–765. https://doi.org/10.1109/tnn.2003.810608

[19] Ozawa S, Roy A, Roussinov D (2009) A multitask learning model for online pattern recognition. IEEE Trans Neural Netw 20(3):430–445. https://doi.org/10.1109/tnn.2008.2007961

[20] Zhuang FZ, Luo P, He Q, et al (2015) Survey on transfer learning research. J Softw 26(1):26–39. https://doi.org/10.13328/j.cnki.jos.004631

[21] Long MS (2014) Transfer learning: Problems and methods. PhD thesis, Department of Computer Science and Technology, Tsinghua University, URL http://ise.thss.tsinghua.edu.cn/%7Emlong/doc/phd-thesis-mingsheng-long.pdf

[22] Day O, Khoshgoftaar TM (2017) A survey on heterogeneous transfer learning. J Big Data 4:1–42. https://doi.org/10.1186/s40537-017-0089-0

[23] Pan SJ, Kwok JT, Yang Q, et al (2008) Transfer learning via dimensionality reduction. In: AAAI, pp 677–682, URL https://www.cse.ust.hk/~qyang/Docs/2008/AAAIsinnoA.pdf

[24] Suykens JA (2008) Data visualization and dimensionality reduction using kernel maps with a reference point. IEEE Trans Neural Netw 19(9):1501–1517. https://doi.org/10.1109/tnn.2008.2000807

[25] Li J, Lu K, Huang Z, et al (2018) Heterogeneous domain adaptation through progressive alignment. IEEE Trans Neural Netw Learn Syst 30(5):1381–1391. https://doi.org/10.1109/tnnls.2018.2868854

[26] Hearst MA, Dumais ST, Osuna E, et al (1998) Support vector machines. IEEE Intell Syst App 13(4):18–28. https://doi.org/10.1109/5254.708428

[27] Huang S, Cai N, Pacheco PP, et al (2018) Applications of support vector machine (svm) learning in cancer genomics. Cancer Genom Proteom 15(1):41–51. https://doi.org/10.21873/cgp.20063

[28] Ribeiro MT, Singh S, Guestrin C (2016) "why should i trust you?" explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, pp 1135–1144, https://doi.org/10.1145/2939672.2939778

[29] Lin Y, Cai Z, Jiang Y, et al (2010) Perioperative risk factors for pulmonary complications after liver transplantation. J Int Med Res 38(5):1845–1855. https://doi.org/10.1177/147323001003800532

[30] Kim J, Nguyen TT, Li Y, et al (2020) Contrasting effects of stored allogeneic red blood cells and their supernatants on permeability and inflammatory responses in human pulmonary endothelial cells. Am J Physiol-lung C 318(3):L533–L548. https://doi.org/10.1164/ajrccm-conference.2020.201.1_meetingabstracts.a5582

[31] Tan L, Wei X, Yue J, et al (2021) Impact of perioperative massive transfusion on long term outcomes of liver transplantation: a retrospective cohort study. Int J Med Sci 18(16):3780. https://doi.org/10.7150/ijms.61697

[32] Benson AB, Burton Jr JR, Austin GL, et al (2011) Differential effects of plasma and red blood cell transfusions on acute lung injury and infection risk following liver transplantation. Liver Transpl 17(2):149–158. https://doi.org/10.1002/lt.22212

[33] Feltracco P, Carollo C, Barbieri S, et al (2013) Early respiratory complications after liver transplantation. World J Gastroentero 19(48):9271. https://doi.org/10.3748/wjg.v19.i48.9271

[34] Pippard B, Bhatnagar M, McNeill L, et al (2022) Hepatic hydrothorax: A narrative review. Pulm Ther 8(3):241–254. https://doi.org/10.1007/s41030-022-00195-8

[35] Hiroi K, Matsusaki T, Kaku R, et al (2019) Postoperative course of serum albumin levels and organ dysfunction after liver transplantation. Transpl P 51(8):2750–2754. https://doi.org/10.1016/j.transproceed.2019.01.199