REVIEW ARTICLE OPEN

Check for updates

Optimization of diabetes prediction methods based on combinatorial balancing algorithm

HuiZhi Shao[1,2], Xiang Liu[2], DaShuai Zong[2] and QingJun Song[2]✉

BACKGROUND: Diabetes, as a significant disease affecting public health, requires early detection for effective management and intervention. However, imbalanced datasets pose a challenge to accurate diabetes prediction. This imbalance often results in models performing poorly in predicting minority classes, affecting overall diagnostic performance.
OBJECTIVES: To address this issue, this study employs a combination of Synthetic Minority Over-sampling Technique (SMOTE) and Random Under-Sampling (RUS) for data balancing and uses Optuna for hyperparameter optimization of machine learning models. This approach aims to fill the gap in current research concerning data balancing and model optimization, thereby improving prediction accuracy and computational efficiency.
METHODS: First, the study uses SMOTE and RUS methods to process the imbalanced diabetes dataset, balancing the data distribution. Then, Optuna is utilized to optimize the hyperparameters of the LightGBM model to enhance its performance. During the experiment, the effectiveness of the proposed methods is evaluated by comparing the training results of the dataset before and after balancing.
RESULTS: The experimental results show that the enhanced LightGBM-Optuna model improves the accuracy from 97.07% to 97.11%, and the precision from 97.17% to 98.99%. The time required for a single search is only 2.5 seconds. These results demonstrate the superiority of the proposed method in handling imbalanced datasets and optimizing model performance.
CONCLUSIONS: The study indicates that combining SMOTE and RUS data balancing algorithms with Optuna for hyperparameter optimization can effectively enhance machine learning models, especially in dealing with imbalanced datasets for diabetes prediction.

Nutrition and Diabetes (2024)14:63 ; https://doi.org/10.1038/s41387-024-00324-z

INTRODUCTION

Diabetes is a major global health issue, projected to affect ~48% of the population by 2045 [1]. This disease arises from pancreatic dysfunction, which may result from insufficient insulin production or inadequate cellular response to insulin, thereby impacting various bodily functions. Alarmingly, about 30 to 50% of diabetes patients worldwide remain undiagnosed [2], underscoring the critical need for predictive diagnosis. Early predictive diagnosis is essential for timely intervention, treatment, and disease management, as it can help control disease progression and reduce complications. Therefore, researching predictive diagnostic methods and technologies for diabetes is crucial. Timely identification of high-risk individuals can significantly improve treatment outcomes and enhance overall quality of life.

El-Hafeez et al. [3] demonstrate the efficacy of machine learning in identifying synergistic combinations of FDA-approved cancer drugs, offering novel approaches to cancer treatment. In addressing the global challenge of Hepatitis C Virus (HCV) prediction, Farghaly et al. [4] employed machine learning techniques on real-world data from Egypt, showcasing the potential of such methods in early disease detection and management. In exploring the progress in disease prediction, CNN robot learning and Grey Wolf Optimizer heuristic optimization algorithm were used to classify monkeypox skin lesions, which can diagnose monkeypox skin lesions faster and more accurately, which has important implications for controlling and preventing monkeypox outbreaks [5]. Omar and El-Hafeez [6] introduce a new approach to enhance seizure recognition, emphasizing the importance of optimizing model performance through deep learning, which includes advanced preprocessing techniques such as feature scaling and discard layers for more accurate diagnosis and treatment of epilepsy. In a recent study by Hady and El-Hafeez [7], machine learning models were developed to predict pelvic tilt and lumbar angle in women experiencing urinary incontinence and sexual dysfunction, highlighting the potential of AI in enhancing diagnostic accuracy and treatment strategies for pelvic floor dysfunctions. In addressing the challenge of class imbalance in cyberbullying datasets, Khairy, Mahmoud, and Abd-El-Hafeez [8] explore various oversampling and undersampling techniques to enhance classification algorithms' performance. Automatic hyperparameter tuning greatly improves the performance and versatility of the model [9]. Recent research by Hassan, El-Hafeez, and Shams [10] delves into optimizing disease classification through advanced language model analysis of

[1]Jinan Engineering Polytechnic, Ji-Nan, Shandong, China. [2]College of Intelligent Equipment, Shandong University of Science & Technology, Tai-an, Shandong, China. ✉email: skdsqj@sdust.edu.cn

Received: 13 October 2023 Revised: 22 July 2024 Accepted: 26 July 2024
Published online: 14 August 2024

symptoms, utilizing Medical Concept Normalization-Bidirectional Encoder Representations from Transformers (MCN-BERT) models and a Bidirectional Long Short-Term Memory (BiLSTM) model, each optimized with distinct hyperparameter optimization methods to predict diseases from symptom descriptions. Sneha and Gangil [11] determined that the decision tree algorithm and random forest are the best classification methods for analyzing diabetes data by leveraging the important feature attributes of the diabetes dataset. Bej et al. [12] proposed a novel method for identifying and characterizing sub-populations of Type-2 diabetes patients in India using an unsupervised machine learning approach, revealing significant heterogeneity in socio-demographic and lifestyle characteristics. Thomas et al. [13] proposed a novel method for identifying and characterizing sub-populations of Type-2 diabetes patients in India using an unsupervised machine learning approach, revealing significant heterogeneity in socio-demographic and lifestyle characteristics.

LightGBM is a gradient boosting decision tree framework that is widely used in machine learning for handling large-scale imbalanced datasets [14, 15]. Wang et al. [16] enhanced diabetes mellitus early warning and factor analysis accuracy by employing ensemble Bayesian networks coupled with SMOTE-ENN and Boruta algorithms. Bakry et al. [17] developed a framework called automatic suppression based on XGBoost for anti-money laundering (ASXAML), enhancing the detection of financial crimes by reducing false positives through the integration of recursive feature elimination with cross-validation and hyperparameter tuning using the Optuna framework. Feng et al. [18] proposed an enhanced coarse aggregate shape classification method leveraging the Per-Optuna-LightGBM model, demonstrating improved accuracy and efficiency in classifying aggregate shapes. Gu et al. [19] introduced a 5CV-Optuna-LightGBM regression prediction model for data prediction, achieving a 99.433% accuracy rate, which demonstrated superior prediction accuracy, higher modeling efficiency, and better fitting compared to other models.

Currently, despite the existence of various diabetes prediction methods, enhancing prediction accuracy, handling imbalanced datasets, and achieving early predictive diagnosis remain important research challenges in the field. Existing literature on diabetes prediction primarily focuses on improving the accuracy of prediction models, with less attention given to the issue of imbalanced datasets. This imbalance, where one class has significantly fewer samples than the other, can lead to inadequate predictive performance for the minority class. Although some researchers have attempted to address the imbalance issue by combining different classifiers or employing data resampling techniques, efficiently integrating data preprocessing and model training processes while maintaining high prediction accuracy and low computational costs remains a critical challenge.

Since the number of diabetic patients is significantly smaller than that of non-diabetic patients, the diabetes prediction problem is modeled as an imbalanced binary classification problem from a machine-learning perspective. First, a combination of SMOTE oversampling and RUS undersampling is used to address the class imbalance in the diabetes dataset, allowing the model to better learn the features of the data. Then, Optuna is employed to automatically search for the optimal hyperparameter combination for the LightGBM model to achieve better predictive performance. Finally, the LightGBM algorithm is used to train the dataset, improving the prediction accuracy for diabetic patients. The rest of this article is organized as follows: Section 2 introduces the mathematical description, architecture, and methods of the diabetes prediction model. Section 3 covers data preprocessing, including data cleaning, data splitting, and data augmentation. Section 4 presents the experiments conducted on the dataset, providing performance metrics evaluation, analyzing the predictive effectiveness of the proposed method, and comparing the proposed method with other methods. Finally, Section 5 offers a brief conclusion.

SUBJECTS AND METHODS

In this article, we combined the efficient learning algorithm of LightGBM with the automated hyperparameter optimization capability of Optuna to form a diabetes prediction framework. This section mainly introduces the mathematical description, architecture, and methods of this model.

Mathematical description

Diabetes prediction can be viewed as a binary classification problem. Given a dataset $D = \{(x_i, y_i)\}_{i=1}^{N}$, where x_i is the feature vector of the i^{th} sample, and $y_i \in \{0, 1\}$ is the corresponding label. The goal is to train a model $f(x)$ to predict the label of unseen samples.

LightGBM Model. LightGBM is a tree-based learning algorithm that optimizes many aspects of traditional Gradient Boosting Decision Trees (GBDT) [20]. LightGBM, when handling large-scale datasets and complex models, can train models faster and more efficiently than traditional GBDT algorithms while maintaining high prediction accuracy. The prediction value $f(x)$ of a LightGBM model can be represented as the weighted sum of multiple decision trees:

$$f(x) = \sum_{k=1}^{K} w_k h_k(x) \tag{1}$$

where K is the number of decision trees, $h_k(x)$ is the prediction result of the k^{th} decision tree, and w_k is the weight of the k^{th} tree.

The key to LightGBM lies in its optimized approach to data handling and decision tree construction. It employs a histogram-based algorithm to accelerate the training process and reduce memory consumption, and uses a leaf-wise growth strategy to enhance model accuracy.

Optuna hyperparameter optimization. Optuna is an automated hyperparameter optimization framework, whose core is to search for the optimal parameter combination by defining an objective function to minimize or maximize its value [21]. In the LightGBM model, Optuna can be used to find the optimal hyperparameters, such as num_leaves, max_depth, learning_rate, and others. In the Optuna-LightGBM-based diabetes prediction model, the hyperparameter combination with the highest cross-validation score is selected.

$$best\_params = argmax\ CV_{Score(params)} \tag{2}$$

Where $best\_params$ is the parameter set that maximizes the cross-validation score CV_{Score}. $CV_{Score(params)}$ is the cross-validation accuracy score given a parameter set. The *params* represents the parameter set of the model, including but not limited to num_leaves, max_depth, and learning_rate.

During the optimization process of Optuna, it automatically tests multiple different parameter combinations. Each set of parameters is evaluated based on its cross-validation score, and the evaluation results determine which set of parameters can provide the best model performance. This process is based on the results of previous experiments to guide the direction of subsequent searches so that the optimal parameter configuration is to be found in the shortest possible time.

Model architecture

In this article, the *diabetes prediction dataset* from 'kaggle' was used, which contains 9 attributes and 100,000 records. The detailed description of the dataset is provided in Table 1.

The nine attributes are gender, age, hypertension status, history of heart disease, smoking history, BMI, glycated hemoglobin level, blood sugar level, and the output result. The output result

Table 1. Data description of the diabetes prediction dataset.

Dataset properties	Descriptions	Value range
Gender	Gender of subjects	Male/Female
Age	Age of subjects	0.08–80
Hypertension	Prevalence of hypertension	1/0
Heart_disease	History of heart disease	1/0
Smoking_history	Smoking history of the subjects	Never/Ever/Former/Not current/Current/No info
Bmi	Body mass index (BMI)	10.01–95.69
HbA1c_level	Glycated hemoglobin level	3.5–9
Blood_glucose_level	Blood glucose levels	80–300

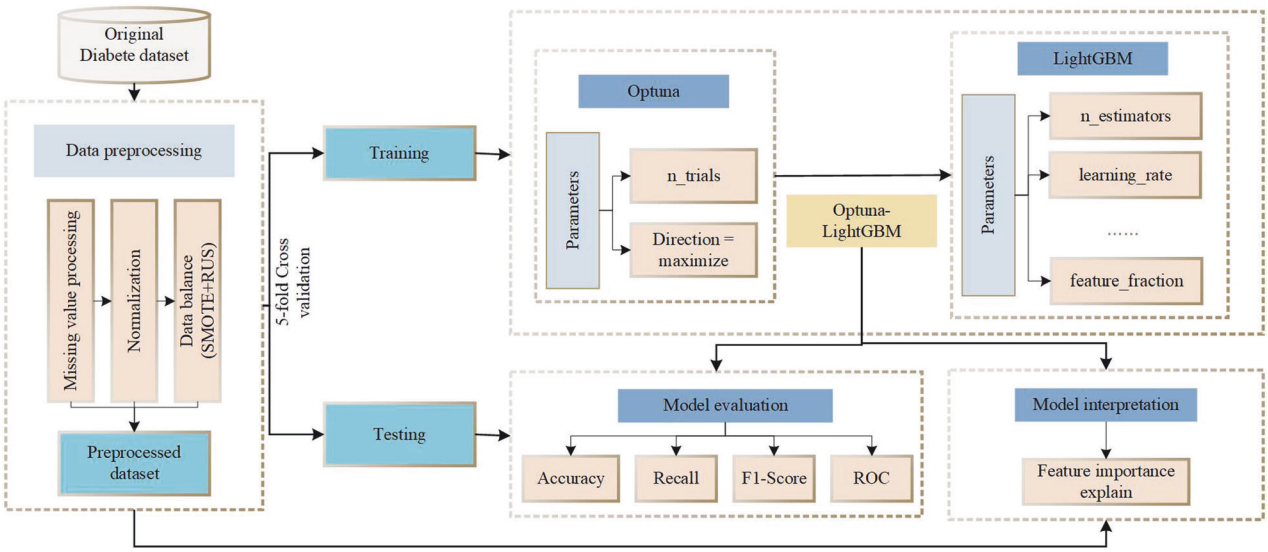

Fig. 1 Overall design of the proposed model.

represents the diabetes status of each individual data point, while the other variables are independent feature variables. Among the 100,000 samples, there are 8,500 patients with diabetes and 91,500 patients without diabetes. The proportion of patients with diabetes is only 8.5%, significantly lower than the 91.5% of non-diabetic patients, making the dataset highly imbalanced.

For effective improvement of the prediction performance of the unbalanced diabetes prediction dataset, Optuna is used to optimize the LightGBM performance parameters, The detailed process of Optuna-LightGBM model is depicted in Fig. 1.

In the Optuna-LightGBM-based diabetes prediction model, data preprocessing, which includes data cleaning, feature engineering (feature selection, feature transformation, etc.), and missing-value processing, is performed first to ensure data quality and improve model performance. Then, data resampling technique (SMOTE + RUS) is used to deal with the data imbalance. Next, a hyperparameter search is performed to automate the search for the optimal hyperparameters of the LightGBM model with Optuna, a step that is achieved by defining the search space, the optimization objective and the cross-validation strategy. Finally, model training, which uses Optuna-optimized hyperparameter settings to train the model with the LightGBM algorithm, and evaluates the model performance with cross-validation methods to select the optimal Optuna-LightGBM model for prediction.

The proposed method of Optuna-LightGBM

When utilizing Optuna for hyperparameter optimization, the first step is to define the search range for each hyperparameter, such as setting the range of values for parameters like learning rate and maximum tree depth. Next, an objective function is designed to measure the performance of the model given the hyperparameters by evaluating the performance metric, accuracy, on the validation set. Through this target function, Optuna can explore the hyperparameter space, iteratively trying different parameter combinations, and adjusting its search strategy based on performance feedback to ultimately determine the best hyperparameter configuration that optimizes model performance. LightGBM parameter settings are shown in Table 2.

When training the LightGBM model, the model is initialized according to the optimal parameter settings provided by Optuna. To solve the problem of data imbalance, the SMOTE + RUS balancing technique was adopted for resampling the training data to improve the data distribution during model training. Subsequently, the LightGBM model was trained on the adjusted data, and a series of decision trees were constructed step by step using the gradient boosting technique to minimize the prediction error. Finally, the performance of the model is evaluated through cross-validation, which uses metrics such as accuracy, recall and F1 score to assess the prediction accuracy of the model.

The proposed model's workflow diagram is shown in Fig. 2. The sharing of parameters between different modules in the diagram enables the model to learn global features better, thereby improving model prediction efficiency and generalization ability. With the aforementioned methods, the diabetes prediction model based on Optuna-LightGBM achieves high accuracy in predicting diabetes risk, which provides a strong support for clinical decision-making.

Table 2. Parameters setting of the LightGBM.

Parameter	Meaning	Range of settings	Values after Optuna optimization
n_estimators	Number of boosting estimators (trees).	50–200	142
learning_rate	Boosting learning rate.	0.01–0.3	0.015
num_leaves	Maximum number of leaves per tree.	2–50	12
max_depth	Maximum depth of each tree.	3–12	5
min_data_in_leaf	Minimum number of data points in a leaf node.	10–100	23
min_child_weight	The minimum sum of weights of all observations required in a child (leaf).	0.001–1	0.0965
subsample	The subsample ratio of the training instance.	0.6–1.0	0.6504
colsample_bytree	The subsample ratio of columns when constructing each tree.	0.6–1.0	0.7115
min_child_samples	Specifies the minimum number of samples (or observations) which are required in a leaf node.	5–100	34
reg_alpha	L1regularization term.	0.001–10.0	0.2165
reg_lambda	L2 regularization term.	0.001–10.0	0.0431
min_split_gain	The minimum gain to perform a split.	0.001–1.0	0.1653
bagging_fraction	Fraction of data to be used for bagging.	0.6–1.0	0.7096
subsample_freq	The frequency for bagging.	1–10	8
bagging_freq	Specifies the frequency for bagging.	1–10	7

The algorithm flow for balancing the dataset using SMOTE and random under-sampling, followed by training and optimizing the LightGBM model with Optuna, is given as follows.

Algorithm 1. Pseudocode of the Improved LightGBM.

Data preprocessing

In machine learning, the quality of input data has a significant impact on the output results, and it is necessary to preprocess the existing dataset before making predictions. Therefore, we need to clean the data to avoid inaccurate prediction results and reduce the misclassification rate of the predictive model. Furthermore, it is also essential to analyze the distribution of the dataset and process the unbalanced dataset through data balancing algorithms that increase the model's perception of the few categories of data and improve its robustness.

Data cleansing. Data cleansing, a crucial step in the data preprocessing phase, is instrumental in achieving data integrity. It involves addressing missing values, eliminating duplicates, and correcting inconsistencies or errors. With it we can significantly reduce the likelihood of biased or inaccurate results in our analyses.

An initial check for missing values in the dataset was performed. By carefully evaluating the presence of "No info" values and making decisions regarding predictors with a high proportion of missing values, we aim to maintain the integrity and quality of the dataset. This approach ensures more reliable analysis and accurate interpretation. The distribution of the cleaned dataset is shown in Fig. 3.

Figure 3 presents the kernel density distribution plots for each attribute of the diseased and non-diseased categories along the main diagonal. The upper triangle shows scatter plots of each attribute's distribution for the two categories, while the lower triangle displays marginal kernel density plots (contour plots) for the attributes of the two categories. It can be observed that in the cleaned dataset, the kernel density distribution trends for diabetic and non-diabetic samples are quite consistent for attributes such as age, hypertension, heart disease, and BMI. However, the distribution differences are more pronounced for HbA1c levels and glucose levels. In the scatter plots and marginal kernel density plots, there is a certain degree of overlap between the samples of the two categories, but overall, the boundaries of the distribution areas for the two categories are relatively clear. This indicates good separability, which is beneficial for training and predicting with the diabetes prediction model.

Data splitting. The performance of machine learning models largely depends on data quality and data strategies [22]. Therefore, it is important to evaluate the impact of data splitting on the performance of machine learning models. The data splitting methods used in this article include K-fold cross-validation and random splitting using the train-test split method.

K-fold cross-validation is a statistical technique used to evaluate the performance of machine learning models. This method divides the dataset into K equally sized subsets. Among these K subsets, each subset is used in turn as the test set, while the remaining K-1 subsets are combined to form the training set for model training. This process is repeated K times, with each iteration selecting a different subset as the test set and the rest as the training set. Ultimately, performance evaluation results for K models are obtained. This process iterates based on the number of folds. The model's generalization performance is estimated by averaging the obtained scores [23], as shown in Fig. 4a.

The train-test split method divides the dataset into random training and testing subsets. This approach depends on the size of the dataset [24], as shown in Fig. 4b.

Correlation analysis. A correlation matrix is used to analyze the relationships among various attributes in the cleaned dataset by statistically calculating the connections or relationships between two or more variables in the dataset. This relationship is measured numerically, with higher values indicating a closer relationship between the inputs and desired outputs. The correlation matrix of the diabetes prediction dataset is shown in Fig. 5.

From the correlation matrix heatmap in Fig. 5, it can be seen that HbA1c levels, glucose levels, and age have a closer relationship with the output results in the cleaned dataset. In the diabetes prediction dataset, the attributes do not show a clear tendency to covary with each other, and there is no high correlation of strong covariance. Additionally, a low correlation of a feature only means that the feature is not useful in the presence of other features, but it does not imply that the feature is unimportant for predicting

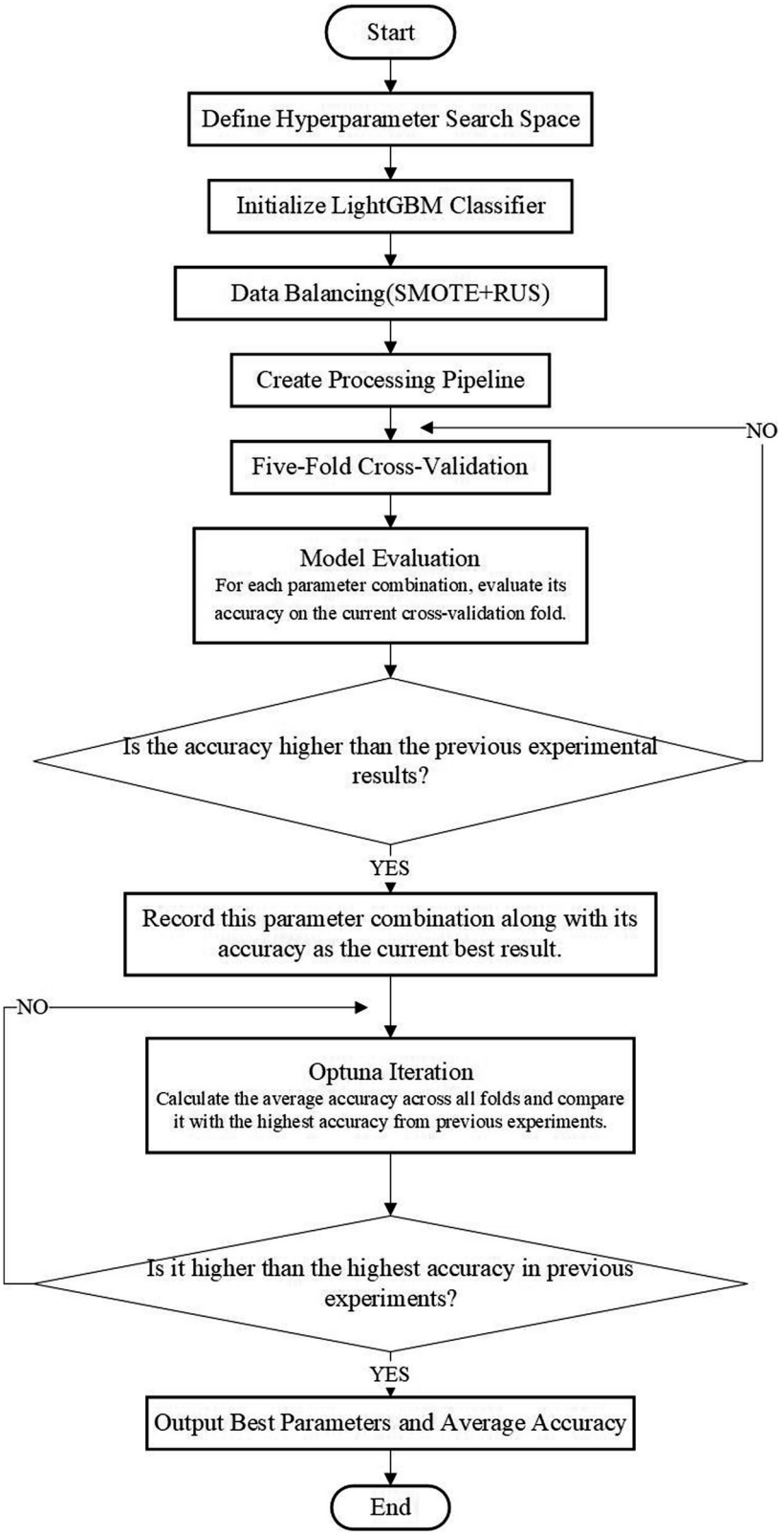

Fig. 2 Flowchart of the proposed method for prediction.

diabetes. In conclusion, the predictors in the diabetes prediction dataset did not show a significant correlation. Therefore, there is no need to worry about high correlations affecting model performance or introducing bias.

Data augmentation. In the diabetes prediction dataset, the data for diabetic patients accounts for only 8.5%, which is <10%, indicating a highly imbalanced state. To balance the class imbalance in the existing dataset, a combination of SMOTE

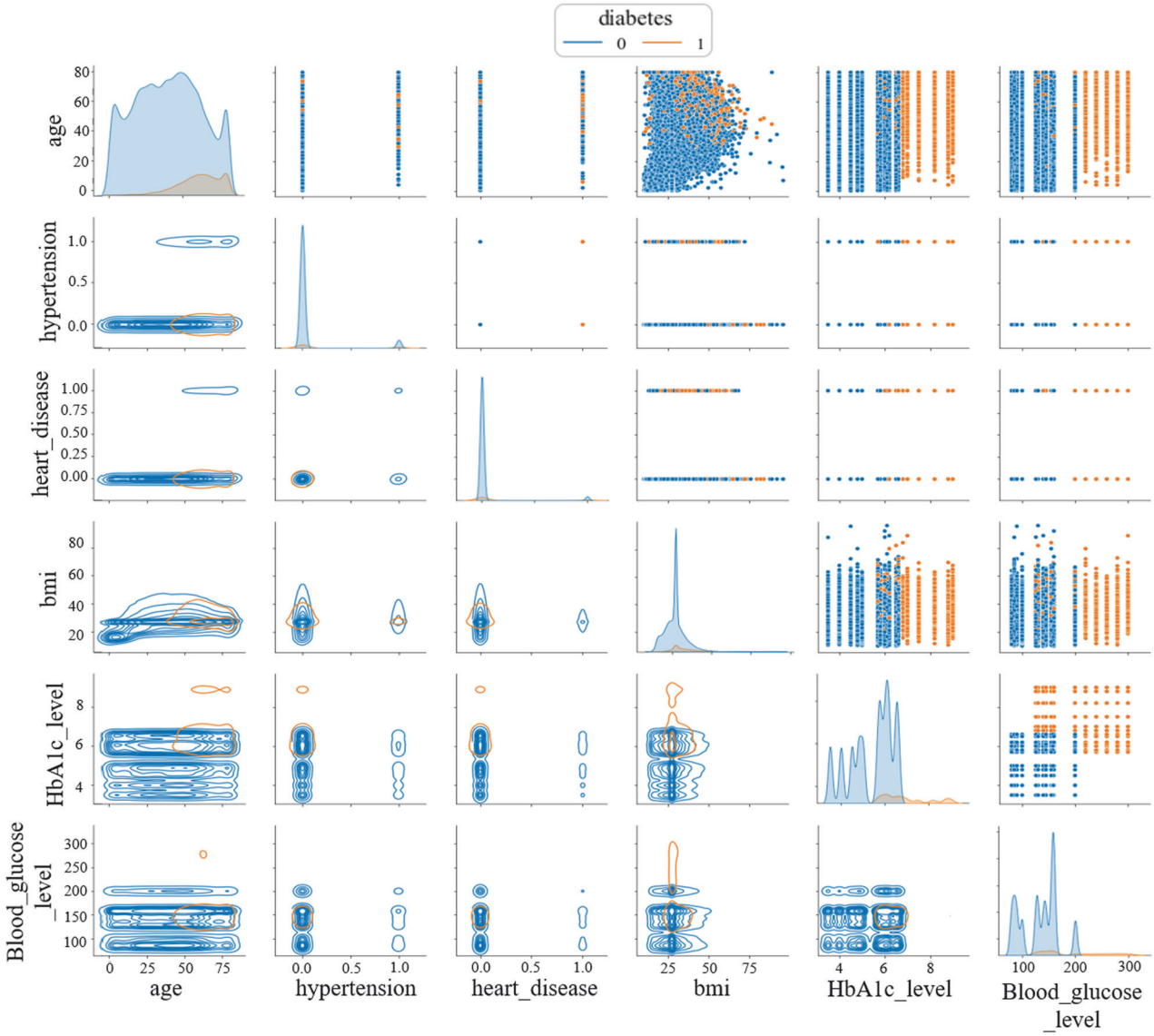

Fig. 3 Distribution of cleaned datasets.

oversampling and RUS undersampling methods is used.

The SMOTE algorithm is an improved solution based on the random oversampling algorithm proposed by Chawla [25], which is a method for oversampling synthetic minority class samples. Since the random oversampling technique simply duplicates samples to increase the minority class samples, it may lead to overfitting issues, where the model learns overly specific information and lacks generalization. The basic idea of the SMOTE algorithm is to analyze the minority class samples and synthetically generate new samples to be added to the dataset.

In an imbalanced dataset, where the set of minority class samples is denoted as $D_{minority}$, the core idea of SMOTE is to interpolate between minority class samples to augment those with fewer data points, thereby balancing the number of samples for different labels. For each sample x_i in $D_{minority}$, the k-nearest neighbors method is used to find a sample x_j.Then, for each pair of x_i and x_j, the difference vector $d = x_i – x_i$ is calculated, and a new sample x_{new} is generated. The expression for x_{new} is as follows:

$$x_{new} = x_i + rand(0,1) \times (x_j - x_i) \tag{3}$$

where $rand(0,1)$ represents a random number within the range (0,1).

The SMOTE algorithm synthesizes new minority class samples to increase the quantity of minority class samples, retaining the information of the original data. However, it does not process the majority class samples, which may result in the loss of some majority class information. The RUS Random undersampling refers to the process of removing some samples from the majority class to reduce redundancy and achieve data balance.

If the set representing the majority class samples is denoted as $D_{majority}$, then the process involves calculating the number of majority class samples to retain, denoted as M, typically approaching or equaling the quantity of minority class samples. Subsequently, M samples are randomly selected from $D_{majority}$.

The combined use of SMOTE and RUS involves first applying SMOTE to increase the number of minority class samples and then performing RUS to decrease the number of majority class samples. Assuming N is the desired number of new minority class samples, the objective of this combined method is to create a new dataset where the number of samples in the minority and majority classes is more balanced.

Fig. 4 Data splitting methods. a Performing k-fold cross-validation. **b** The dataset is divided into two parts randomly with different ratios: 80: 20, 70:30, and 60:40 train/test split.

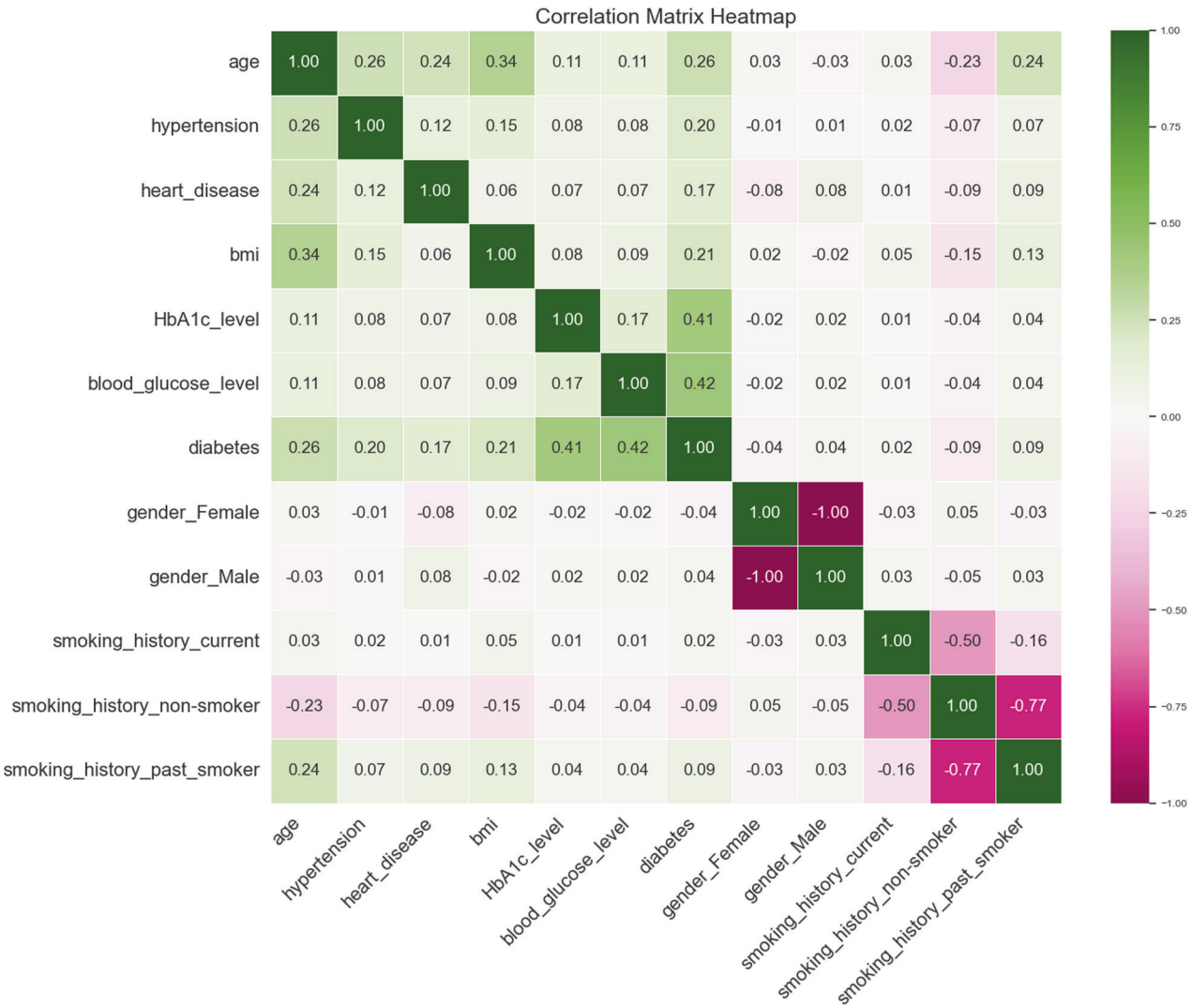

Fig. 5 Correlation matrix heat map.

The process of combining SMOTE and RUS can be outlined in the following steps:

1. Apply SMOTE to generate new minority class samples $D'_{minority}$ until the quantity reaches N.
2. Perform random undersampling from the majority class $D_{majority}$ to obtain $D'_{majority}$.

The final synthesized dataset $D_{combined}$ is obtained by merging $D'_{minority}$ and $D'_{majority}$':

$$D_{combined} = D'_{minority} \bigcup D'_{majority} \tag{4}$$

This combined approach allows fine-tuning of the class balance of the new dataset by adjusting N and M.

Evaluation index

Given that the accuracy of predicting diabetic patients has a greater impact than that of non-diabetic patients, this article uses the confusion matrix and the receiver operating characteristic (ROC) curve [26] as experimental evaluation metrics. Note that in diabetes prediction, more attention is paid to the accuracy of predicting diabetic patients, aiming to minimize false negatives (predicting diseased as healthy). The evaluation metrics from the confusion matrix are listed in Table 3.

The data in the confusion matrix are used to estimate a set of statistically relevant performance indicators defined as follows.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \tag{5}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{6}$$

$$Recall = \frac{TP}{TP + FN} \tag{7}$$

$$\text{F1} - \text{Score} = \frac{2TP}{2TP + FP + FN} \tag{8}$$

In the ROC curve, the true positive rate (TPR) is plotted on the y-axis and the false positive rate (FPR) is plotted on the x-axis.

Table 3. Model evaluation based on binary confusion matrix.

Confusion matrix		Actual value Positive	Negative
Recognition value	Positive	True positive (TP)	False positive (FP)
	Negative	False negative (FN)	True negative (TN)

The TPR is mathematically the same as the recall rate, whereas the FPR indicates how many false positives occurred out of all the available negative samples during the testing period. The formulas for calculating the TPR and FPR are as follows:

$$\text{Truepositiverate} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{9}$$

$$\text{Falsepositiverate} = \frac{\text{FP}}{\text{FP} + \text{TN}} \tag{10}$$

RESULTS

This section includes four main experiments: data splitting experiments, enhancement experiments with combined balancing algorithms, model optimization comparison experiments, and comparative experiments of different machine learning models on the diabetes prediction dataset.

Data splitting experiments

Data splitting methods used in this article are k-fold cross-validation and random splitting using train–test splits method. Using the random splitting method, the dataset was divided into two parts, with different ratios: 70:30 and 60:40 train/test split, in contrast performing k-fold cross-validation using $k = 5$ and $k = 10$. We have utilized all the features for different data partitioning methods for diabetes dataset on Multi-layer Perceptron (MLP) [27], Support Vector Machines (SVM) [28], Decision Tree [29, 30], Random Forest (RF) [31, 32], eXtreme Gradient Boosting (XGBoost) [33], and LightGBM on which comparative analyses were performed.

Table 4 shows that different classifiers exhibit high accuracy across various data splitting methods, with LightGBM and MLP performing the best, consistently achieving accuracy above 96%. The accuracy is not sensitive to changes in the training and testing split ratios, indicating that the dataset is well-representative. Additionally, the accuracy difference between 5-fold and 10-fold cross-validation is minimal. For example, LightGBM achieves 97.05% accuracy with 5-fold cross-validation and 97.06% with 10-fold cross-validation, demonstrating high model stability. Therefore, 5-fold cross-validation is chosen as the data splitting method. The reason for choosing 5-fold cross-validation is that it provides a good balance between computational efficiency and model stability and is sufficient when the dataset size is not exceptionally large.

Combined balancing algorithm effect enhancement experiment

Based on the optimal data splitting method obtained from the data splitting experiments, this article chooses to use 5-fold cross-validation. The dataset is balanced using SMOTE, RUS, and a combination of SMOTE + RUS, respectively, along with

Table 4. Comparison of different classifiers using two data splitting methods in terms of accuracy.

Classifier	Data splitting approach			K-fold cross-validation	
	Train-test splits				
	80:20 (%)	70:30 (%)	60:40 (%)	$K = 5$ (%)	$K = 10$ (%)
RF	96.54	96.63	96.7	96.81	96.77
LightGBM	96.94	96.91	97.03	97.05	97.06
XGBoost	96.91	96.85	96.97	97.04	97
DecisionTree	94.73	94.78	94.78	95.03	95.03
MLP	96.96	96.92	97.03	97.05	97.04
SVM	96	95.96	96.07	96.18	96.19

Table 5. Comparison of accuracy using different class balancing methods.

Classifier	Unbalanced	SMOTE	RUS	SMOTE + RUS
RF	96.81	95.7	92.66	96.41
LightGBM	97.05	97	93.53	97.07
XGBoost	97.04	96.95	93.45	97.01
DecisionTree	95.03	94.49	91.09	94.66
MLP	97.05	90.63	89.95	95.95
SVM	96.18	88.45	88.54	95.65

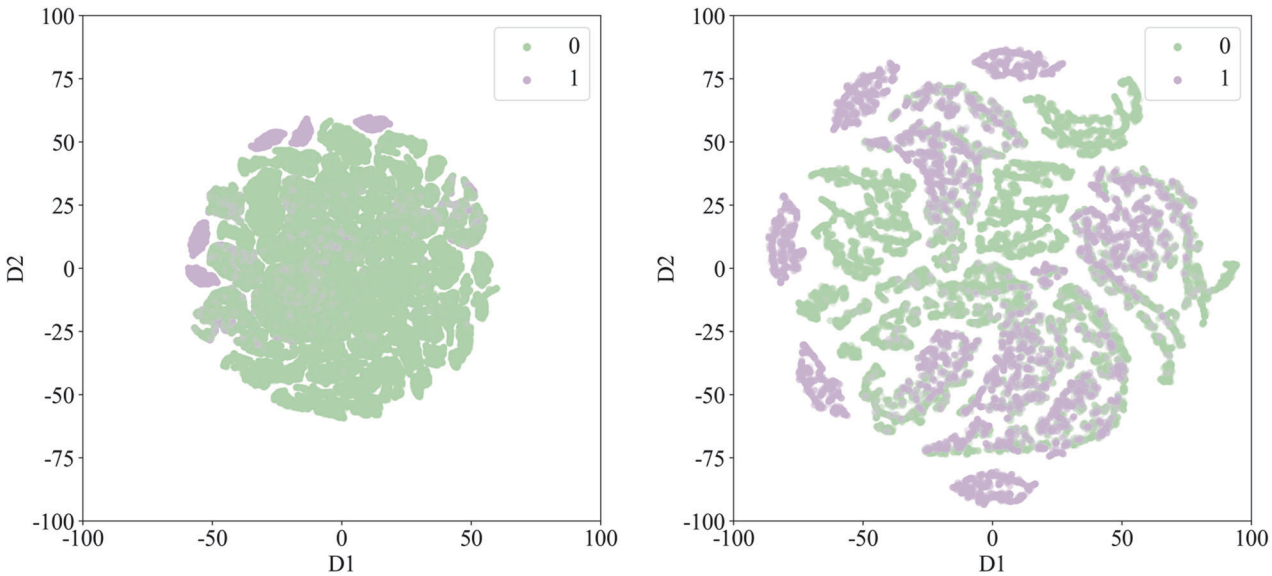

Fig. 6 Scatterplot of t-SNE dimensionality reduction of the dataset before and after treatment with the combined equilibrium approach.

comparisons to the unbalanced data in terms of the accuracy of the six machine learning models. The experimental results are shown in Table 5.

According to the experimental results, the LightGBM classifier demonstrates high performance on both unbalanced and balanced datasets using SMOTE, RUS, and SMOTE + RUS methods. By comparing different balancing methods, we found that the SMOTE + RUS method significantly improves the accuracy of the LightGBM classifier. Specifically, the accuracy of LightGBM under the SMOTE + RUS method reaches 97.07%, an increase of 0.02% compared to the unbalanced condition. This indicates that the SMOTE + RUS method can achieve a better balance between the majority and minority classes in the LightGBM classifier, thereby enhancing the model's generalization ability. Therefore, SMOTE + RUS is chosen as the data balancing method.

To illustrate the usefulness of the combined SMOTE + random downsampling balancing algorithm proposed in this article and the performance of the classifier model on the balanced dataset, t-SNE visualization technique [34] was applied to the datasets before and after the balancing process. The t-SNE scatter plots of the datasets in both states are shown in Fig. 6, where label 0 represents the samples without diabetes and label 1 represents the samples with diabetes.

It can be seen that the original data without balanced processing has a small proportion of the diseased category and the distance between the data points of the two categories is small, and there is also more overlap, which makes the separability poorer. On the dataset processed by the combined balancing algorithm, the proportion of data points of the two categories tends to be balanced, the imbalance of the dataset is improved, and the distance between the data points of the two categories is further compared to that of the unbalanced dataset, with enhanced separability.

Model optimization comparison experiments

In this section, we explore the role of the Optuna search method for model hyperparameter optimization and compare it with two traditional methods, random search and grid search. In random search, hyperparameter values are randomly selected and iterated over a specified number of search times. In grid search, a range of discrete values for each hyperparameter is specified, and these values are uniformly divided and combined. Each possible combination is then tried one by one. The average results of different parameter optimization methods on the standard diabetes prediction dataset using 5-fold cross-validation under the LightGBM model are shown in Table 6.

Table 6 shows that only the algorithm proposed in this article achieves improvements in both accuracy and precision compared to the unoptimized model, with the highest values obtained. The Accuracy is increased by 0.04% to 97.11%, and the Precision is increased by 1.82% to 98.99%. This indicates that the proposed Optuna-LightGBM model can dynamically adjust the search space based on the search history of each hyperparameter, and intelligently explore potential optimal combinations. As a result, it achieves more precise model construction on the data, leading to higher prediction accuracy and effectiveness. In contrast, the random search and grid search methods slightly underperform the non-optimized model in terms of accuracy and precision. This is because they do not consider prior knowledge or experience and only sample hyperparameters randomly within the search space, potentially overlooking some hyperparameter combinations with latent advantages. In terms of time consumption for a

Table 6. Performance of optimization methods.

	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Time(s)
No optimization model	97.07	97.17	68.82	80.57	—
Random search optimization	97.02	96.28	68.95	80.35	46
Grid search optimization	96.97	94.94	69.35	80.15	1788
The algorithm in this article	97.11	98.99	68.03	80.55	2.5

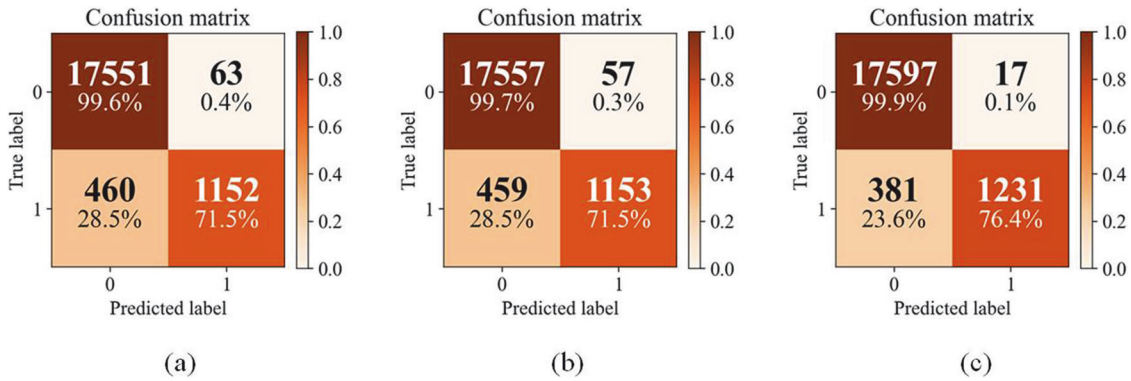

Fig. 7 Confusion matrix of the model under three search methods. a Random search, **b** grid search, **c** Optuna.

single search, grid search requires 1788 seconds, significantly longer than other optimization methods. This is due to its inherent characteristic of needing to exhaustively explore all possible hyperparameter combinations. Consequently, as the number of hyperparameters and their value ranges increase, the time required grows exponentially. In the case of large or complex search space, grid search is less applicable. The single search time of the proposed model in this article is only 2.5 s, which is the fastest among all the compared methods. This is because the Optuna algorithm takes advantage of Bayesian optimization, which determines the next hyperparameter selection in each iteration based on prior knowledge and search history, and estimates the potential maximum values of different hyperparameter values for the objective function by modeling the available data and basing the selection on the expectation of the different hyperparameters. This probability-based approach can effectively utilize the available information to improve the accuracy of the next hyperparameter selection, thereby accelerating convergence and improving search efficiency.

Figure 7 shows the confusion matrices of the prediction results for the three search methods. The numbers above the confusion matrix represent the predicted sample quantities, while the values below represent the recall percentages. As can be seen from the figure, the confusion matrix obtained after the optimization of the algorithm in this article achieves 99.9% and 76.4% recall on non-diabetic and diabetic respectively, which are higher than the other two methods. Among them, the recall on the samples of the diabetes category improves more, which is 4.9% higher than both the random search and the grid search, which indicates that the model has better prediction results for the diabetes category. The comparison results reflect that the Optuna-LightGBM model proposed in this article can intelligently adjust the search space and effectively utilize the existing information, which is able to mine the data features more deeply and reduce the probability of incorrect prediction.

Comparative experiments of various machine learning algorithms

In this section, we combine five classic machine learning models—MLP, SVM, Decision Tree, RF, and XGBoost—with the Optuna algorithm and compare them with the proposed Optuna-LightGBM model to validate its prediction and generalization capabilities. The average test results for the six models are shown in Table 7. Here, 'Before' represents the performance of the models before Optuna hyperparameter optimization, and 'Optimized' represents their performance after optimization.

From the comparison of the six classifier models in Table 7, it can be seen that most models show improved performance after Optuna optimization, particularly in terms of accuracy and F1 score. This indicates that the optimization strategy enables the models to better capture key features in the data, thereby enhancing their generalization ability. For example, the accuracy of the Decision Tree model increased from 94.66% to 97.08% after optimization, and the Average F1-Score increased by 8.94%. This significant improvement is attributed to Optuna's capability to optimize hyperparameters. By effectively exploring the hyperparameter space using Bayesian optimization, Optuna identifies the optimal hyperparameter combinations, significantly enhancing the performance of the Decision Tree model. Decision Tree models are particularly sensitive to hyperparameters such as tree depth and minimum samples per split, which directly affect model complexity and generalization ability.

In contrast, although the XGBoost algorithm performs superiorly in a number of tasks, it slightly underperforms the LightGBM algorithm in terms of hyper-parameter optimization efficiency and handling large-scale data. The Optuna-LightGBM model proposed in this article exhibits the highest average accuracy of 97.11% after optimization, as well as small improvements in recall and F1-Score, and a single search time of only 2.5 seconds, showing a good balance of efficiency and performance. This improvement in accuracy is attributed to LightGBM's inherent ability to handle large-scale data and high-dimensional features. By accelerating training through a histogram-based decision tree algorithm, LightGBM's key parameters are finely tuned, allowing the model to better capture complex patterns in the data while avoiding overfitting. This balance of efficiency and performance demonstrates that the Optuna-LightGBM model can ensure high prediction accuracy while significantly reducing training time, making it an efficient and superior choice for handling this dataset.

To further demonstrate the superiority of the proposed model's performance, ROC curves of the six models were analyzed, as

Table 7. Comparison results before and after optimization of each model.

Model		Accuracy (%)	Recall (%)	F1-score (%)	Time (s)
MLP	Before	95.95	75.71	76.76	–
	Optimized	96.03	76.99	76.3	23.5
SVM	Before	95.65	72.21	74.57	–
	Optimized	96.04	74.42	76.85	171.2
Decision tree	Before	94.66	74.58	71.19	–
	Optimized	97.08	66.86	80.13	1.3
RF	Before	96.41	71.12	77.66	–
	Optimized	97.08	66.86	80.13	52.5
XGBoost	Before	97.01	69.26	80.33	–
	Optimized	97.08	66.99	80.18	9.4
The algorithm in this article	Before	97.07	68.82	80.57	–
	Optimized	97.11	68.03	80.55	2.5

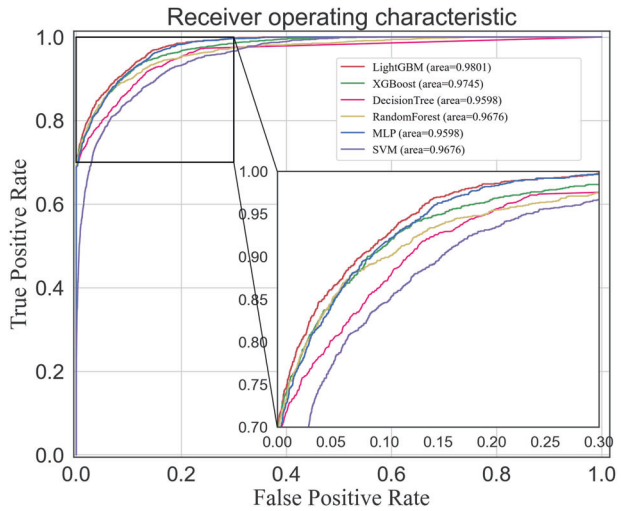

Fig. 8 ROC curves for four machine learning models.

shown in Fig. 8. Among the area under the curve of each model in the figure, the area under the curve of the Optuna-LightGBM algorithm is the largest, reaching 0.9801, and the curve is closest to the upper left corner. This indicates that the LightGBM algorithm can achieve a higher rate of true cases under the same threshold while maintaining a lower rate of false positive cases. There are several reasons for the excellent performance of the LightGBM algorithm. Firstly, LightGBM adopts a histogram-based decision tree algorithm, which reduces computational complexity and improves split-point selection accuracy by discretising continuous feature values, so that the relationship between the features and the target variables can be captured more accurately. Secondly, the depth of the tree and the number of leaf nodes are finely tuned by Optuna algorithm optimization to ensure that the model can capture complex feature interactions while avoiding overfitting, thus optimizing the true and false positive rates. In addition, the LightGBM algorithm introduces gradient-based one-sided sampling (GOSS), which selects high-gradient samples for splitting, improves computational efficiency, ensures attention to key samples, and enhances the ability of minority class identification. These advantages enable the Optuna-LightGBM model proposed in this article to capture the complex features of the data while maintaining excellent generalization ability, which creates its excellent performance in terms of AUC and overall performance.

Feature importance analysis

In this section, we analyze feature importance in detail under the Optuna-LightGBM model to understand which features play a key role in diabetes prediction. Feature importance reflects how much each feature influences the model decision and is an effective way to assess the contribution of features.

The LightGBM algorithm measures feature importance by counting the number of times a feature is used for segmentation in the tree structure. Whenever a feature is used to segment the data, it receives an importance score that accumulates over time. In this way, we can visualize which features are used frequently in the model and infer their importance.

In Fig. 9, HbA1c levels stand out as a critical predictor with an importance score of 0.496, significantly higher than other features. HbA1c provides a comprehensive assessment of an individual's blood sugar levels over the past 2 to 3 months, placing it at the core of diabetes risk evaluation. Following closely is blood glucose level, with an importance score of 0.301, further confirming the crucial role of glucose levels in diabetes diagnosis. Age has an importance score of 0.282, highlighting the association between age and the risk of type 2 diabetes, which is consistent with medical research. The BMI importance score is 0.108, indicating that body mass index also plays an important role in diabetes risk assessment. Although hypertension and heart disease have lower importance scores, they still show some relevance, suggesting that these factors may play a role in diabetes prediction.

Feature importance analysis indicates that HbA1c levels, glucose levels, age, and BMI are the most critical features in diabetes prediction. These findings not only align with medical knowledge but also enhance our understanding of the model's decision-making process, helping to further optimize the model and improve prediction accuracy and reliability. In practical applications, these features can be monitored as key indicators to facilitate early detection and intervention for diabetes.

DISCUSSION

As a major disease affecting public health, early detection of diabetes is crucial. In this article, a data balancing method combining SMOTE and RUS is proposed to address the category imbalance problem of diabetes dataset, which effectively mitigates the negative impact of category imbalance on model performance. And the improved LightGBM-Optuna model is used to optimize the prediction performance. Experimental results show that the method exhibits better prediction accuracy and efficiency than the traditional method on the unbalanced diabetes prediction dataset.

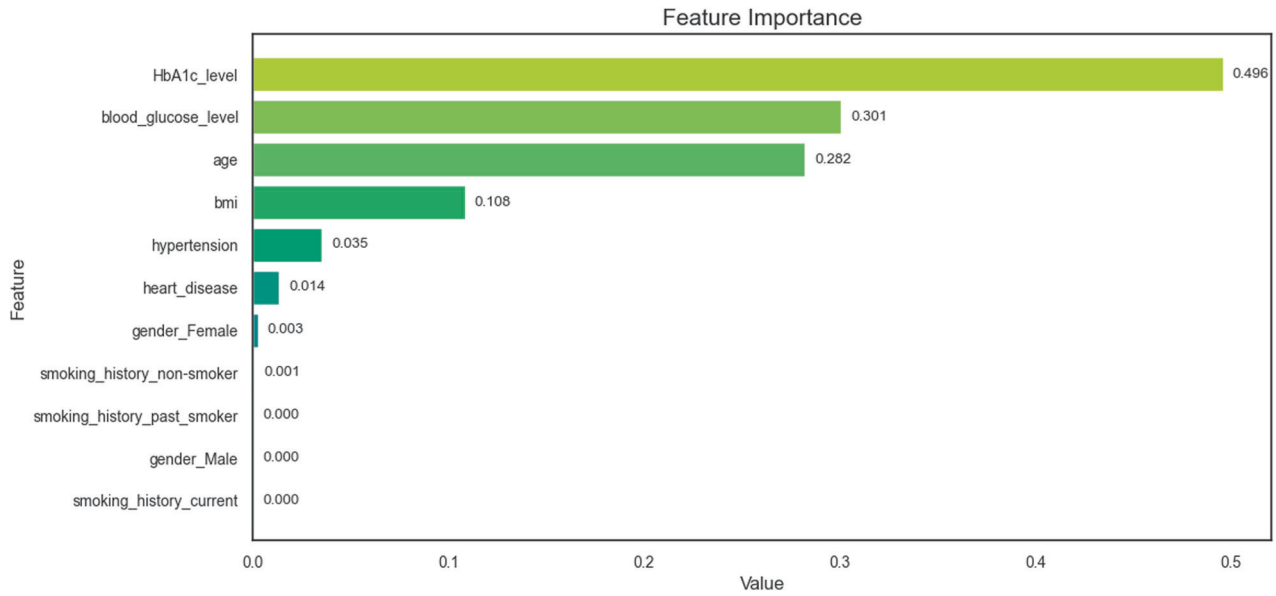

Fig. 9 Feature importance ranking results.

The experimental results demonstrate the superior performance of the Optuna-LightGBM algorithm in terms of accuracy, time efficiency and generalization capability. Specifically, the accuracy improves from 97.07% to 97.11%, an improvement of 0.04, the precision improves from 97.17% to 98.99%, an improvement of 1.82, and the time for a single search is only 2.5 seconds, which is much faster than that of the traditional grid search method. In addition, the model showed superior adaptability and robustness across different datasets and conditions, and the feature importance analysis was consistent with medical knowledge, highlighting the importance of HbA1c levels and blood glucose levels for diabetes prediction.

Despite the encouraging results, this study has some limitations. Firstly, training and validation were based on the current dataset, which may suffer from selection bias and does not fully reflect the characteristics of all patients. Second, although the model performed well on the current dataset, its generalization ability still needs to be validated in a larger population and diverse clinical settings. In addition, the model may not fully capture all potential risk factors, especially those not recorded in the dataset. Future studies should consider more predictive variables, such as lifestyle and dietary habits, to further improve prediction accuracy. Finally, model interpretability still needs to be improved to gain trust and application by healthcare practitioners.

Overall, the Optuna-LightGBM model shows potential for early diabetes prediction, but further research is needed to overcome current limitations and validate the model's effectiveness and feasibility. Future studies could explore methods to integrate more diverse data sources and validate the model's generalization ability across different populations, as well as develop more transparent and interpretable models.

DATA AVAILABILITY

The data used to support the findings of this article are available from the corresponding author upon request.

REFERENCES

1. Standl E, Khunti K, Hansen TB, Schnell O. The global epidemics of diabetes in the 21st century: Current situation and perspectives. Eur J Prev Cardiol. 2019;26:7–14. https://doi.org/10.1177/2047487319881021.
2. Makungu Marvellous Chauke. Possible correlations between HbA1c and selected modifiable risk factors for type 2 diabetes mellitus in a non-diabetic population. 2022. http://hdl.handle.net/102000/0002.
3. Abd El-Hafeez T, Shams MY, Elshaier YAMM, Farghaly HM, Hassanien AE. Harnessing machine learning to find synergistic combinations for FDA-approved cancer drugs. Sci Rep. 2024;14. https://doi.org/10.1038/s41598-024-52814-w.
4. Mamdouh Farghaly H, Shams MY, Abd El-Hafeez T. Hepatitis C Virus prediction based on machine learning framework: a real-world case study in Egypt. Knowl Inf Syst. 2023;65:2595–617. https://doi.org/10.1007/s10115-023-01851-4.
5. Eliwa EHI, El Koshiry AM, Abd El-Hafeez T, Farghaly HM. Utilizing convolutional neural networks to classify monkeypox skin lesions. Sci Rep. 2023;13. https://doi.org/10.1038/s41598-023-41545-z.
6. Omar A, Abd El-Hafeez T. Optimizing epileptic seizure recognition performance with feature scaling and dropout layers. Neural Comput Appl. 2024;36:2835–52. https://doi.org/10.1007/s00521-023-09204-6.
7. Abdel Hady DA, Abd El-Hafeez T. Predicting female pelvic tilt and lumbar angle using machine learning in case of urinary incontinence and sexual dysfunction. Sci Rep. 2023;13. https://doi.org/10.1038/s41598-023-44964-0.
8. Marwa K, Mahmoud TM, Abd-El-Hafeez T. The effect of rebalancing techniques on the classification performance in cyberbullying datasets. Abstract Neural Comput Appl. 2024;36:1049–1065. https://doi.org/10.1007/s00521-023-09084-w.
9. Mahabub A. A robust voting approach for diabetes prediction using traditional machine learning techniques. SN Appl Sci. 2019;1. https://doi.org/10.1007/s42452-019-1759-7.
10. Hassan E, Abd El-Hafeez T, Shams MY. Optimizing classification of diseases through language model analysis of symptoms. Sci Rep. 2024;14. https://doi.org/10.1038/s41598-024-51615-5.
11. Sneha N, Gangil T. Analysis of diabetes mellitus for early prediction using optimal features selection. J Big Data. 2019;6. https://doi.org/10.1186/s40537-019-0175-6.
12. Bej S, Sarkar J, Biswas S, Mitra P, Chakrabarti P, Wolkenhauer O. Identification and epidemiological characterization of Type-2 diabetes sub-population using an unsupervised machine learning approach. Nutr Diabetes. 2022;12. https://doi.org/10.1038/s41387-022-00206-2.
13. Thomas DM, Kleinberg S, Brown AW, Crow M, Bastian ND, Reisweber N, et al. Machine learning modeling practices to support the principles of AI and ethics in nutrition research. Nutr Diabetes. 2022;12. https://doi.org/10.1038/s41387-022-00226-y.
14. Hajihosseinlou M, Maghsoudi A, Ghezelbash R. A novel scheme for mapping of MVT-type Pb–Zn prospectivity: lightGBM, a highly efficient gradient boosting decision tree machine learning algorithm. Nat Resour Res. 2023;32:2417–38. https://doi.org/10.1007/s11053-023-10249-6.
15. Zhang Y, Yu W, Li Z, Raza S, Cao H. Detecting ethereum ponzi schemes based on improved lightGBM algorithm. IEEE Trans Comput Soc Syst. 2022;9:624–37. https://doi.org/10.1109/TCSS.2021.3088145.

16. Wang X, Ren J, Ren H, Song W, Qiao Y, Zhao Y, et al. Diabetes mellitus early warning and factor analysis using ensemble Bayesian networks with SMOTE-ENN and Boruta. Sci Rep. 2023;13. https://doi.org/10.1038/s41598-023-40036-5.
17. Bakry AN, Alsharkawy AS, Farag MS, Raslan KR. Automatic suppression of false positive alerts in anti-money laundering systems using machine learning. J Supercomput. 2024;80:6264–84. https://doi.org/10.1007/s11227-023-05708-z.
18. Feng X, Sun Z, Xing Z, Wu Y, Lian C. Coarse aggregate shape classification method based on per-optuna-lightGBM model. J Phys: Conf Ser. 2023;2589:12015. https://doi.org/10.1088/1742-6596/2589/1/012015.
19. Liang GU, Zinan T, Rong J. Data prediction method based on 5CV-optuna-lightGBM regression model. Softw. Eng. 2024;27:49–54. https://doi.org/10.19644/j.cnki.issn2096.
20. Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, et al. LightGBM: a highly efficient gradient boosting decision tree. Neural Inf Process Syst. 2017; 3149–57. https://github.com/Microsoft/LightGBM.
21. Akiba T, Sano S, Yanase T, Ohta T, Koyama M. Optuna: a next-generation hyperparameter optimization framework. knowledge discovery and data mining. 2019; 2623–31. http://arxiv.org/abs/1907.10902.
22. Asteris PG, Apostolopoulou M, Armaghani DJ, Cavaleri L, Chountalas AT, Guney D, et al. On the metaheuristic models for the prediction of cement-metakaolin mortars compressive strength. Metaheuristic Comput Appl. 2020;1:63–99. https://doi.org/10.12989/mca.2020.1.1.063.
23. Kohavi R. A study of cross-validation and bootstrap for accuracy estimation and model selection. Int Jt Conf Artif Intell. 1995;2:1137–43. http://robotics.stanford.edu/~ronnyk.
24. Liu H, Cocea M. Semi-random partitioning of data into training and test sets in granular computing context. Granul Comput. 2017;2:357–86. https://doi.org/10.1007/s41066-017-0049-2.
25. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: synthetic minority over-sampling technique. J Artif Intell Res. 2002;16:321–57.
26. Qin Z-C. Roc Analysis For Predictions Made By Probabilistic Classifiers. International Conference on Machine Learning and Cybernetics. 2005. p. 3119–24.
27. Zhen Z. Research on key techniques of cardiovascular diseases risk prediction based on machine learning. 2021; https://link.cnki.net/doi/10.27005/d.cnki.gdzku.2021.003835.
28. Jegan C, Kumari VA, Chitra R. Classification of diabetes disease using support vector machine. Int J Eng Res Appl. 2013;3:1797–801. https://www.researchgate.net/publication/320395340.
29. Quinlan JR, Rivest RL. Inferring decision trees using the minimum description length principle. Inf Comput. 1989;80:227–48.
30. Agrawal R, Ghosh S, Imielinski T, Iyer B. An interval classifier for database mining applications. Proceedings of the 18th International Conference on Very Large Data Bases. 1992.
31. Lee JW, Lee JB, Park M, Song SH. An extensive comparison of recent classification tools applied to microarray data. Comput Stat Data Anal. 2005;48:869–85. https://doi.org/10.1016/j.csda.2004.03.017.
32. Yeung KY, Bumgarner RE, Raftery AE. Bayesian model averaging: development of an improved multi-class, gene selection and classification tool for microarray data. Bioinformatics. 2005;21:2394–402. https://doi.org/10.1093/bioinformatics/bti319.
33. Chen T, Guestrin C. XGBoost: a scalable tree boosting system. Knowl Discov Data Min. 2016;785–94. https://doi.org/10.1145/2939672.2939785.
34. Nl LVU, Hinton G. Visualizing data using t-SNE laurens van der Maaten. J Machine Learn Res. 2008;9:2579–605.

ACKNOWLEDGEMENTS

This work was supported by National Natural Science Foundation of China (52174145), Innovation capability improvement project of scientific and technological small and medium-sized enterprises of Shandong Province China (2022TSGC1271, 2023TSGC0620), Science and Technology Innovation Department Project of Tai'an City Shandong Province China (2021ZDZX006).

AUTHOR CONTRIBUTIONS

HS led the study design, developed the methodology, and drafted the initial manuscript. XL developed the software tools for analysis, validated the results, and contributed to manuscript review and editing. DZ curated the data, conducted a formal analysis, and visualized the results. QS supervised the project, acquired funding, and managed project administration to ensure the research objectives were met.

COMPETING INTERESTS

The authors declare no competing interests.

ADDITIONAL INFORMATION

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41387-024-00324-z.

Correspondence and requests for materials should be addressed to QingJun Song.

Reprints and permission information is available at http://www.nature.com/reprints

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.