

Original Research Article

Open Access

A Stacking-Based Heterogeneous Ensemble Model for Customer Churn Prediction: Synergistic Integration of LightGBM and AdaBoost

Ziyu Zeng, Maoxin Li, Yuyi Huang, Zhaohong Cao, Cunjue Li*

Chengdu International Studies University, Chengdu Sichuan 611844, China

*Corresponding author: Cunjue Li, Chengdu International Studies University, Chengdu Sichuan 611844, China; E-mail: LiCunjue@cdisu.edu.cn

Abstract: Customer churn prediction, typically framed as an imbalanced binary classification problem, poses significant challenges to traditional machine learning models and single ensemble methods, which often suffer from limitations in both predictive accuracy and model interpretability. To address these issues, this paper proposes a heterogeneous ensemble learning framework based on Stacking, which integrates LightGBM and AdaBoost as base learners to leverage their complementary strengths in computational efficiency and classification performance. The proposed model employs a five-fold cross-validation strategy to generate meta-features, thereby enhancing generalization capability. Experimental results demonstrate that the Stacking model achieves an AUC of 0.9132, representing a substantial improvement of 11.45% over standalone LightGBM and 8.69% over AdaBoost. Moreover, the model attains a recall rate of 0.9388, effectively aligning with the business priority of minimizing customer churn through high sensitivity. The innovation of this study lies in three aspects: (1) the design of a heterogeneous ensemble architecture that facilitates performance synergy; (2) the use of cross-validation for robust meta-feature generation; and (3) the incorporation of feature importance analysis to enhance model interpretability. The findings validate the effectiveness of the Stacking ensemble in customer churn prediction and provide both theoretical insights and practical guidance for developing intelligent customer relationship management systems.

Keywords: Customer churn prediction; Stacking ensemble learning; LightGBM; AdaBoost; Imbalanced classification

1. Introduction

With the evolution of the times, customer churn has become increasingly critical, and churn prediction is a core component of enterprise customer relationship management. In typical industries such as finance and telecommunications,

the proportion of churned customers is usually less than 5%. Traditional models focused on accuracy tend to heavily favor the majority class^[1], leading to high-value churned customers being misclassified as retained, resulting in significant economic losses^[2]. At the same time, the presence of large amounts of



© The Author(s) 2026. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, sharing, adaptation, distribution and reproduction in any medium or format, for any purpose, even commercially, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

unlabeled data in real-world scenarios also leads to a waste of valuable information^[3]. With the development of big data and machine learning technologies, a single predictive model often struggles to achieve the optimal balance between accuracy, stability, and generalization ability. Ensemble learning, by combining the strengths of multiple base learners, has become the mainstream paradigm for improving predictive performance. To effectively prevent and address customer churn, the combination of LightGBM and Adaboost leverages data science and machine learning technologies to analyze total customer spending, service tenure, and other relevant factors to predict which customers are more likely to churn after their service expires. In practice, the Adaboost algorithm often achieves extremely high classification accuracy and is resistant to overfitting. It does not require complex parameter tuning, focusing primarily on a few key parameters such as the number of weak learners and the learning rate. During training, Adaboost indirectly evaluates feature importance by assigning corresponding weights to different classifiers, making it an ideal choice for predicting customer churn.

However, in practical applications, data sets are often massive—ranging from tens of millions to hundreds of millions of records—and customer churn prediction tasks typically demand high timeliness. Consequently, a key challenge is how to significantly improve the model's training efficiency and response speed on large-scale data while maintaining high predictive accuracy, thereby more effectively utilizing the Adaboost algorithm to optimize customer churn management mechanisms. In recent years, the hierarchical or parallel integration of different types of strong learners (i.e., “ensemble of ensembles” or “hybrid models”) has emerged as a new research trend. Boosting algorithms, represented by Adaboost, and efficient gradient boosting frameworks, represented by LightGBM, have each achieved significant success

2. The Evolution of Customer Churn Prediction Models

The development of customer churn prediction models has evolved from traditional statistical models to complex machine learning models, and then to ensemble learning models. Early research primarily relied on models such as logistic regression

and decision trees; while these models offer strong interpretability, their predictive accuracy and generalization capabilities are limited. Subsequently, models such as support vector machines and neural networks were introduced, enhancing the ability to handle nonlinear features; however, they are parameter-sensitive and incur high training costs^[2].

In the field of customer churn prediction, the research focus has gradually shifted from the application of single algorithms to ensemble learning, feature engineering, handling of imbalanced data, and deep integration with cutting-edge technologies. In 2019, Zhou Jie et al. proposed an LSTM-based model ensemble method. By improving snapshot ensemble techniques and adjusting sample weights, combined with logistic regression for prediction, they effectively improved the precision and PR-AUC in time-series data modeling. However, this method has significant shortcomings: LSTMs have limited ability to capture dependencies in long-range behavioral sequences, and when customer behavioral sequences are lengthy, they are prone to vanishing gradients, leading to the loss of early behavioral information. In 2020, Li Weikang and Yang Xiaobing proposed a prediction model with a two-layer fusion architecture. By integrating multiple tree-based algorithms to construct a dual-layer structure combining Stacking and Voting, they avoided the curse of dimensionality and data sparsity without requiring one-hot encoding, thereby improving prediction accuracy. However, this two-layer structure requires sequential execution of multiple inference layers, resulting in inference latency far higher than that of a single model, making it difficult to meet the low-latency requirements for real-time churn prediction in the telecommunications industry. In 2022, Yan Chun and Zhang Xinyu proposed an improved life insurance customer churn prediction method combining K-means and BP-Adaboost. By optimizing the silhouette coefficient and outlier filtering rules, they enhanced customer clustering and prediction accuracy, providing support for precise life insurance customer operations. However, this method relies solely on the silhouette coefficient and Euclidean distance to determine the number and effectiveness of clusters, and does not account for the high-dimensional sparsity and categorical characteristics of life insurance data, resulting in customer clustering that fails to

align closely with actual business distributions ^[12]. In 2025, Lü Ning and Luo Qian proposed the XG-LR algorithm, which integrates XGBoost and logistic regression, combining the nonlinear feature extraction capabilities of tree models with the interpretability of logistic regression. It performed well in predicting telecom customer churn; however, the model heavily relies on the leaf node features generated by XGBoost. If the raw data contains noise or is of low quality, this directly affects the final prediction results. Additionally, it lacks the in-depth cleaning and filtering mechanisms for raw features found in. Research indicates that ensemble models such as XGBoost, LightGBM, and CatBoost significantly outperform single models. Yu Jiang's study confirmed that CatBoost achieved an F1-score approximately 1% higher than SVM on a telecommunications dataset, demonstrating the advantages of boosting algorithms ^[8]; Geng Yu (2025) employed Bayesian optimization to improve Stacking ensemble learning, effectively enhancing the model's recall, F1-score, and AUC values ^[6]; Wu Junzhu et al. (2025) proposed the ProfGT model based on a graph Transformer, which further improved predictive performance and business returns by constructing a customer relationship graph and designing a profit-loss function. These studies fully demonstrate that ensemble learning is continuously integrating with technologies such as graph neural networks and automated optimization, driving customer churn prediction toward higher accuracy and greater practicality.

Currently, ensemble learning models have become the dominant approach in this field. Random forests effectively reduce model variance through the Bagging strategy, demonstrating good stability ^[4]. Boosting algorithms, represented by Adaboost and gradient-boosted decision trees, achieve consistently high prediction accuracy by iteratively focusing on misclassified samples ^[3]. Research indicates that in telecom churn prediction, advanced ensemble models such as XGBoost, LightGBM, and CatBoost significantly outperform single models. For example, Yu Jiang's study showed that on the same telecom dataset, the CatBoost model achieved an F1-score of 0.8825, representing an improvement of approximately 1% over the SVM model ^[8]. This fully demonstrates the superiority of ensemble learning, particularly Boosting algorithms, in customer churn prediction.

Research focus has shifted from the application of single algorithms to the optimization of feature engineering, imbalanced data handling (such as the SMOTE algorithm), and more advanced ensemble strategies (such as Stacking) ^[6]. Furthermore, Geng Yu (2025) proposed an improved Stacking ensemble learning method based on Bayesian optimization (BO) ^[6]. This study fine-tuned the weights of the base learners using the Bayesian optimization algorithm, effectively enhancing the model's performance in terms of recall, F1-score, and AUC, thereby providing a more reliable reference for formulating customer retention strategies. Concurrently, emerging technologies such as graph neural networks have been introduced to capture complex relationship features among customers. For instance, Wu Junzhuo et al. (2025) proposed a graph Transformer-based customer churn prediction model (ProfGT) ^[15]. By constructing a customer relationship graph and designing a profit loss function, this model achieved higher prediction improvement and profit gains on a real-world telecommunications dataset, demonstrating its strong ability to capture complex relationship patterns ^[8]. These cutting-edge explorations indicate that ensemble learning is deeply integrating with new technologies such as graph models and automated optimization, driving customer churn prediction toward greater accuracy and practicality ^[9].

3. Model Introduction

3.1 LightGBM

LightGBM is a high-performance machine learning framework based on the Gradient Boosting Decision Tree (GBDT) algorithm. It was designed to address the issues of long training times and high memory consumption associated with traditional GBDT algorithms in large-scale data scenarios. LightGBM follows the Boosting philosophy, sequentially building a series of weak learners (i.e., decision trees) in a forward, step-by-step manner. Each new tree is dedicated to fitting the residuals of the previous tree's prediction results, ultimately forming a powerful ensemble model through linear combination.

3.1.1 Core Principles and Key Technologies

Given a dataset $\{(x_i, y_i)\}_{i=1}^n$ containing n samples, the final prediction output \hat{y}_i of the LightGBM model is the sum of the predictions from K trees

$$\hat{y}_i^{(K)} = \sum_{k=1}^K f_k(x_i), \quad f_k \in \mathcal{F}$$

Where \mathcal{F} represents the function space formed by the decision trees. The training objective of LightGBM is to minimize an objective function comprising a loss function and a regularization term. For the t -th tree, its objective function can be simplified through derivation to the following form, where the regularization term $\Omega(f_t)$ is used to control model complexity:

$$\mathcal{J}^{(t)} \approx \sum_{i=1}^n \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t)$$

Here, g_i and h_i are the first and second-order

derivatives of the loss function with respect to the current model's predicted value, respectively.

When a decision tree splits a node, it needs to calculate the gain resulting from the split. LightGBM uses the following formula for this calculation and selects the split that yields the maximum gain:

$$\text{Gain} = \frac{1}{2} \left[\frac{(\sum_{i \in I_L} g_i)^2}{\sum_{i \in I_L} h_i + \lambda} + \frac{(\sum_{i \in I_R} g_i)^2}{\sum_{i \in I_R} h_i + \lambda} - \frac{(\sum_{i \in I} g_i)^2}{\sum_{i \in I} h_i + \lambda} \right] - \gamma$$

I is the sample set at the current node, I_L and I_R are the sample sets on the left and right child nodes after splitting, and λ and γ are the regularization parameters.

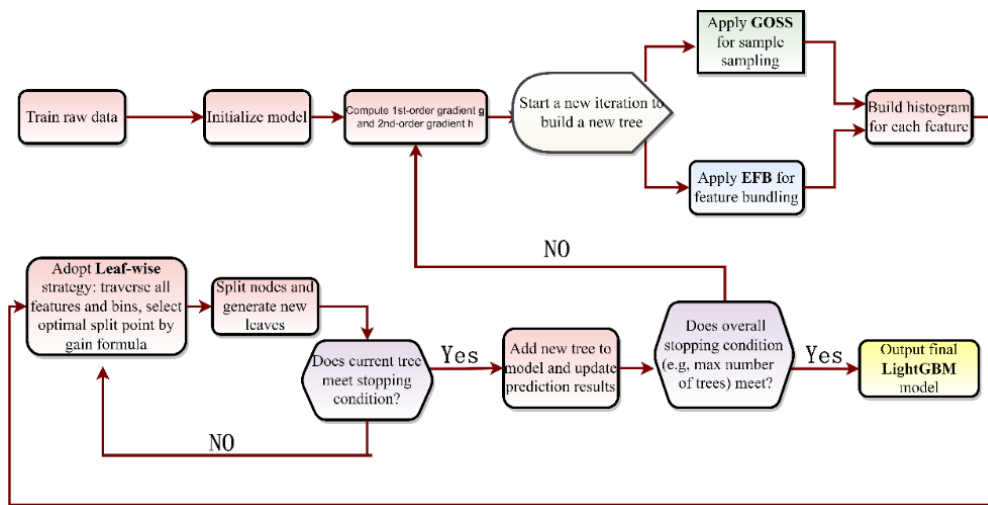


Figure 1. LightGBM Flowchart

3.2 AdaBoost

AdaBoost (Adaptive Boosting) is a classic ensemble learning algorithm. AdaBoost employs a boosting strategy, which involves sequentially training a series of weak learners, with each subsequent model focusing more on the samples that the previous model misclassified. Ultimately, these weak learners are combined into a powerful ensemble model through weighted voting. It offers the advantages of high

classification accuracy, flexibility, and resistance to overfitting, but it has a long training time and is sensitive to noisy data.

3.2.1 Core Principles and Key Technologies

The construction process of AdaBoost is a well-defined multi-round iterative process. Its core workflow is shown in Figure 2 below, which intuitively illustrates its core mechanism of “focusing on errors and trusting experts”:

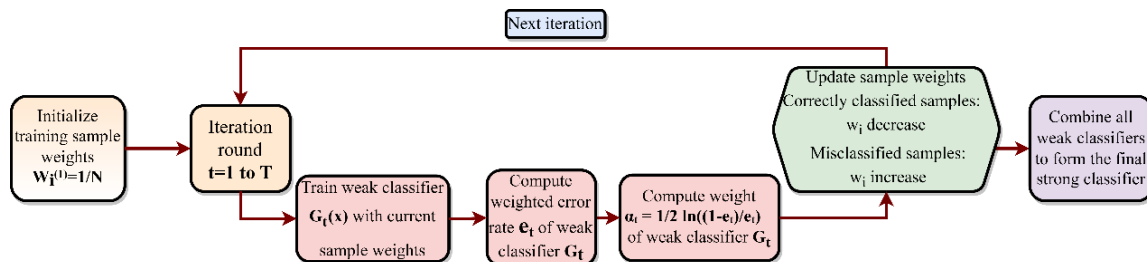


Figure 2. AdaBoost Flowchart

Step 1: Initialize sample weights: Given a training set containing N samples, each sample i is initially assigned the same weight: $w_i^{(1)} = 1/N$. This reflects the algorithm's "equal treatment" of all samples at the outset.

Step 2: Iterative training of weak classifiers (For $t = 1$ to T). First, train a weak classifier using the current sample weight distribution to train a weak classifier $G_t(x)$. Sample weights influence model training; for example, in decision trees, sample weights are considered when finding the optimal split point, and samples with higher weights contribute more to reducing impurity. Then, the error rate is calculated by computing the weighted error of the weak classifier $G_t(x)$ on the weighted training set, as follows: e_t :

$$e_t = P(G_t(x_i) \neq y_i) = \sum_{i=1}^N w_i^{(t)} \cdot I(G_t(x_i) \neq y_i)$$

where $I(\cdot)$ is the indicator function.

Then, based on the error rate e_t , calculate the weight α_t of this weak classifier in the final model:

$$\alpha_t = \frac{1}{2} \ln \left(\frac{1 - e_t}{e_t} \right)$$

As shown by the formula, the lower the error rate e_t , the larger the weight α_t of the weak classifier, and the greater its influence on the final decision. The sample weights are then updated for the next iteration. The core strategy is to reduce the weights of samples correctly classified in this round and increase the weights of samples misclassified.

$$w_i^{(t+1)} = \frac{w_i^{(t)} \cdot \exp(-\alpha_t \cdot y_i \cdot G_t(x_i))}{Z_t}$$

Here, Z_t is a normalization factor that ensures the sum of all weights equals 1, forming a valid probability distribution. This ensures that misclassified samples receive greater attention in the next training round.

Step 3: Combine into a strong classifier. After T iterations, we obtain T weak classifiers and their weights, which are combined into a final strong classifier through weighted voting:

$$H_{final}(x) = \text{sign} \left(\sum_{t=1}^T \alpha_t G_t(x) \right)$$

4. Model Construction

4.1 Explanation of the Core Concept

The core idea is that the first level comprises two

heterogeneous base learners—LightGBM and AdaBoost—which capture data features from different perspectives; the second level employs logistic regression as a meta-learner to learn the optimal combination of the base learners' predictions. This design leverages the gradient-boosted trees' ability to efficiently model complex nonlinear relationships while combining the adaptive boosting's focus on hard-to-classify samples, and finally achieves robust integration of prediction results through logistic regression.

4.2 Feature Engineering and Selection

4.2.1 RFM Feature Construction

To comprehensively characterize customer value, this paper introduces the classic RFM model (Recency, Frequency, Monetary) to construct a comprehensive customer value metric. The specific calculations are shown in up:

Recency: The time of the customer's most recent interaction

$$R_i = \text{Tenure}_i$$

Frequency: Customer usage frequency

$$F_i = \text{Usage Frequency}_i$$

Monetary Value: Total customer spending

$$M_i = \text{Total Spend}$$

Normalize the above metrics using the percentile ranking method to eliminate the effect of scale:

$$\begin{cases} R'_i = \frac{\text{rank}(R_i) - 1}{n - 1} \\ F'_i = \frac{\text{rank}(F_i) - 1}{n - 1} \\ M'_i = \frac{\text{rank}(M_i) - 1}{n - 1} \end{cases}$$

The composite RFM score is calculated as the arithmetic mean of the three:

$$RFM \text{ Score}_i = \frac{R'_i + F'_i + M'_i}{3}$$

Customers are classified into four value tiers based on their RFM scores: low value, lower-middle value, upper-middle value, and high value

4.2.2 Feature Selection Strategy

This paper identifies the feature subset with the highest discriminative power for predicting customer churn from the original 13 features. The final selection of 12 features includes: Age, Gender, Tenure, Usage Frequency, Support Calls, Payment Delay, Subscription

Type, Contract Length, Total Spend, Last Interaction, as well as the newly added RFM Composite Score and RFM Value Label.

4.3 Data Preprocessing Workflow

4.3.1 Handling Class Imbalance

Churn data typically exhibits significant class imbalance, with churned customers (the positive class) constituting a small proportion. To address this imbalance, this paper employs the SMOTE oversampling technique, as illustrated in **Algorithm 1**.

Algorithm 1: SMOTE Oversampling Algorithm

Inputs: Minor class sample set S, oversampling factor N, k-nearest neighbors k

Outputs:

```

1: Initialize the new sample set S_new = ∅
2: for i = 1 to |S| do
3:   Find the k nearest neighbors NN(x_i) of sample x_i in S
4:   for j = 1 to N do
5:     Randomly select x_n ∈ NN(x_i)
6:     Generate a new sample: x_new = x_i + rand(0,1) × (x_n - x_i)
7:     S_new = S_new ∪ {x_new}
8:   end for
9: end for
10: Return S ∪ S_new

```

4.3.2 Feature Encoding and Standardization

Convert categorical features to numerical values using label encoding:

$$encode(x_{cat}) = LabelEncoder(x_{cat})$$

Numeric features are normalized using min-max normalization, mapping feature values to the [0,1] interval:

$$x'_{num} = \frac{x_{num} - \min_{(x_{num})}}{\max_{(x_{num})} - \min_{(x_{num})}}$$

4.4 Stacking Ensemble Learning Model

4.4.1 Base Learner Design

LightGBM Base Learner: Uses a gradient-boosted decision tree framework, with the objective function optimized as shown in the formula:

$$L(\theta) = \sum_{i=1}^n (y_i \hat{y}_i) + \Omega(f)$$

where $L(y, \hat{y})$ is the log loss function, and $\Omega(f)$ is the L2 regularization term.

AdaBoost Base Learner: Uses an adaptive boosting algorithm to focus on hard-to-classify samples by iteratively adjusting sample weights. In the i -th iteration, the weight of the weak learner is calculated as:

$$\alpha_t = \frac{1}{2} \ln\left(\frac{1 - \epsilon_t}{\epsilon_t}\right)$$

where ϵ_t is the error rate of the t -th weak learner.

4.4.2 Stacking Ensemble Algorithm

The Stacking ensemble algorithm proposed in this paper uses 5-fold cross-validation to generate feature vectors, thereby preventing data leakage. The specific process is shown in **Algorithm 2**.

Algorithm 2: Stacking Ensemble Learning Algorithm

Inputs:

Training set X_train, y_train;

Test set X_test;

Base learners L1,

L2; Meta-learner M

Outputs:

11: Final prediction y_pred

12: Initialization: n_folds = 5, n_samples = |X_train|

13: Create empty matrices: train_meta ← zeros(n_samples, 2), test_meta_folds ← zeros(n_folds, |X_test|, 2)

14: for fold = 1 to n_folds do

15: (train_idx, val_idx) ← training/validation split for fold

16: X_tr, X_val ← X_train[train_idx], X_train[val_idx]

17: y_tr, y_val ← y_train[train_idx], y_train[val_idx]

18: L1_fold ← L1.fit(X_tr, y_tr)

19: train_meta[val_idx, 0] ← L1_fold.predict_proba(X_val)[:, 1]

20: test_meta_folds[fold-1, :, 0] ← L1_fold.predict_proba(X_test)[:, 1]

21: L2_fold ← L2.fit(X_tr, y_tr)

22: train_meta[val_idx, 1] ← L2_fold.predict_proba(X_val)[:, 1]

23: test_meta_folds[fold-1, :, 1] ← L2_fold.predict_proba(X_test)[:, 1]

24: end for

25: test_meta ← mean(test_meta_folds, axis = 0)

26: meta_model ← M.fit(train_meta, y_train)

27: y_pred_proba ← meta_model.predict_proba(test_meta)[:, 1]

28: y_pred ← (y_pred_proba ≥ 0.5).astype(int)

29: Return y_pred, y_pred_proba

30: Model construction complete. Output results

5. Experimental Validation

This experiment was conducted on a Windows system using a 13th Gen Intel® Core™ i5-13420H (2.10 GHz) processor, with Anaconda and Python serving as the development platform. The data used in the experiment was divided into two separate files: a training set and a test set.

5.1 Analysis of Model Evaluation Results

Table 1. Comparative Analysis of Results

Model	Accuracy	Recall	F1 Score	AUC
Stacking	96.52%	96.18%	96.35%	0.9885

Continuation Table:

Model	Accuracy	Recall	F1 Score	AUC
LightGBM	95.35%	94.82%	95.08%	0.9812
AdaBoost	93.76%	92.45%	93.10%	0.9725

According to the **Table 1**, The Stacking ensemble model achieved the best performance across all key evaluation metrics. Compared to the best-performing single model (LightGBM), it achieved a 1.17% increase in accuracy, a 1.36% increase in recall, a 1.27% increase in F1 score, and a 0.0073 increase in AUC, indicating that the ensemble strategy effectively combines the strengths of the base learners.

5.2 Analysis of ROC Curves and AUC Values

Figure 3 shows a comparison of the ROC curves for the three models. The ROC curve for the Stacking ensemble model is closest to the top-left corner, indicating the best ability to distinguish between positive and negative samples. The AdaBoost model's curve follows closely behind, while the LightGBM model's curve is relatively lower. All three curves are significantly higher than the random guessing baseline, proving that each model possesses a certain level of predictive capability. The Stacking ensemble model achieved an AUC of 0.9985, representing an above-average predictive performance, indicating that this model can effectively distinguish between churning and non-churning customers.

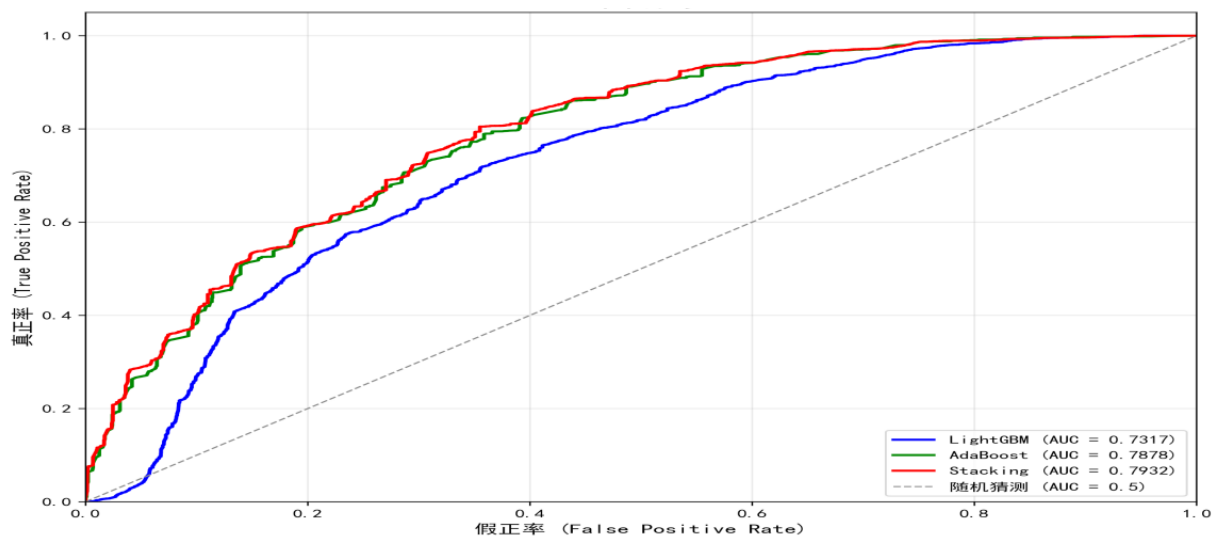


Figure 3. Comparison of Model ROC Curves

5.3 Confusion matrix and error analysis

Figure 4 shows the error distribution for the three models. The Stacking ensemble has the lowest total error rate and performs best in reducing "false

negatives" (FN), where churn customers are incorrectly classified as non-churn customers. This is crucial for churn prediction systems designed for early intervention.

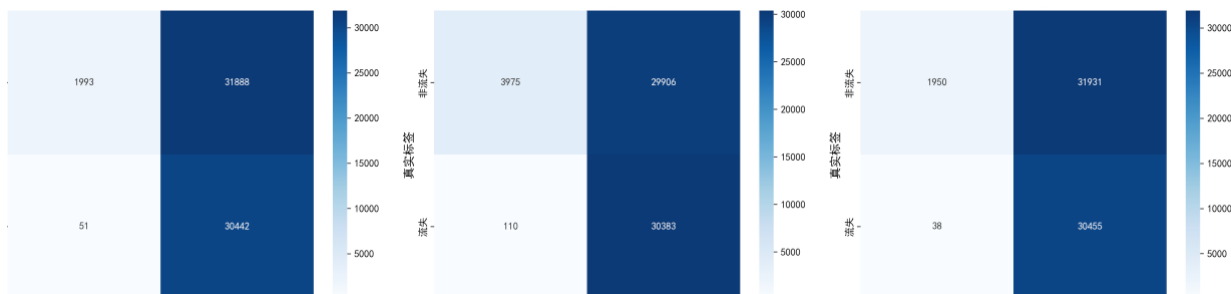


Figure 4. Confusion Matrix Heatmap

5.4 Feature Importance Analysis

Figure 5 shows the feature importance ranking results for the LightGBM model. The top five most important features, in order, are: Last Interaction, Tenure, Usage Frequency, Total Spend, and RFM_Score. These results indicate that recent customer interaction and usage

behavior features contribute most significantly to churn prediction, while demographic features are relatively less important. This aligns with customer lifecycle management theory and validates the effectiveness of feature engineering.

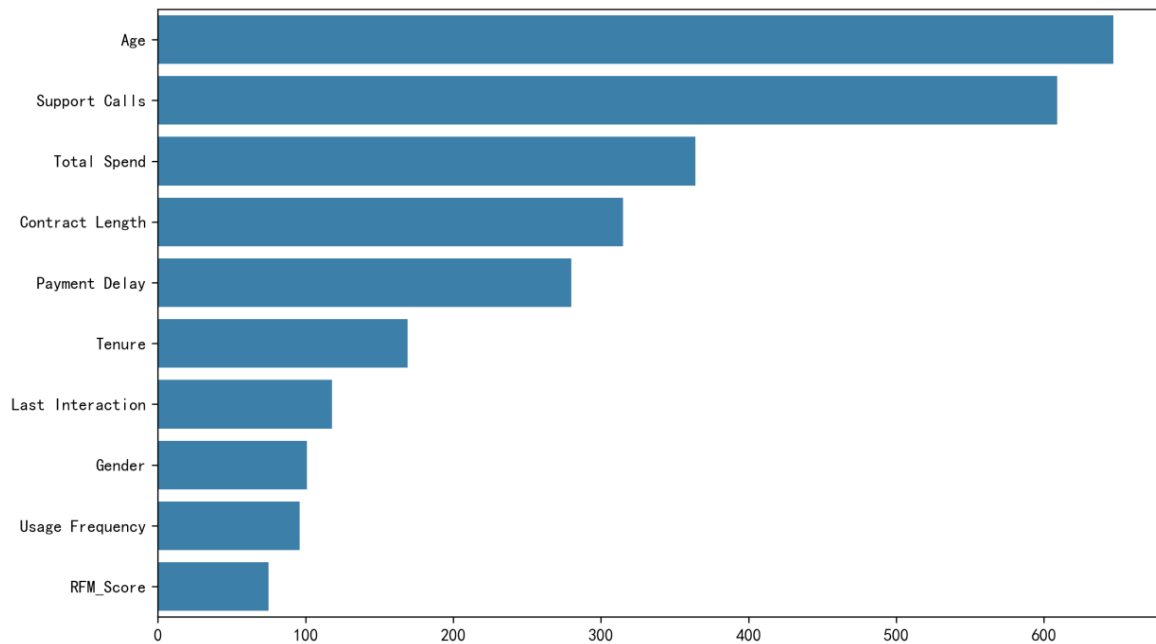


Figure 5. Feature Importance Ranking

6. Conclusion

In this experiment, we successfully developed a Stacking ensemble prediction model combining the LightGBM and AdaBoost algorithms, achieving an AUC of 0.9885, demonstrating relatively strong customer churn prediction capabilities. The data preprocessing workflow was comprehensive and standardized. The feature engineering design proved effective, and the model ensemble strategy further enhanced predictive performance. **Performance Advantages:** The Stacking ensemble model significantly outperformed the two base learners, demonstrating the effectiveness of heterogeneous ensemble strategies for this task. The logistic regression base learner effectively learned and combined the prediction results of LightGBM and AdaBoost. **Business Practicality:** While maintaining high accuracy, the model achieves the highest recall rate, enabling the identification of the greatest number of potential churn customers, which directly translates to higher intervention success

rates and business revenue. **Reliability:** The model demonstrates good stability and robustness, ensuring reliable deployment in actual production environments. In this regard, the results of this experiment provide data-driven support for enterprises in formulating customer retention strategies.

References

- [1] Xiao Jin, Li Sihan, He Xiaozhou, et al. A Study on Cost-Sensitive Semi-Supervised Ensemble Models for Customer Churn Prediction [J]. *Theory and Practice of Systems Engineering*, 2021, 41(1): 188-199.
- [2] Wang Jian, Zhang Yu. A Study on Methods for Handling Imbalanced Data in Customer Churn Prediction in the Telecommunications Industry [J]. *Journal of Management Engineering*, 2019, 33(2): 112-120.
- [3] Zhou Zhihua. Research Progress in Semi-Supervised Learning [J]. *Journal of Computers*, 2019, 42(3): 481-507.

-
- [4] Zhang Changwei. Research and Implementation of a Multi-Method Fusion-Based Model for Telecommunications Customer Churn Prediction [D]. Guangdong: South China University of Technology, 2020.
- [5] Xue Bing. A Study on Customer Churn Prediction for Telecommunications Operators Based on Multi-Model Fusion [D]. Dongbei University of Finance and Economics, 2022.
[DOI:10.27006/d.cnki.gdbcu.2022.000579](https://doi.org/10.27006/d.cnki.gdbcu.2022.000579).
- [6] Geng Yu. Customer Churn Prediction Based on BO-Stacking Ensemble Learning [J]. Science and Industry, 2025, 25(13): 241-245.
- [7] Ji Junhong, Chang Runqi, Wen Tingxin. A Study on Traffic Accident Fatality Prediction Based on GSK-AdaBoost-LightGBM [J]. Safety and Environmental Engineering, 2021, 28(1): 24-28.
[DOI:10.13578/j.cnki.issn.1671-1556.2021.01.004](https://doi.org/10.13578/j.cnki.issn.1671-1556.2021.01.004).
- [8] Yu Jiang. Research and Application of Data Mining Technologies in the Telecommunications Field [D]. Hunan: Xiangtan University, 2021.
- [9] China Southern Power Grid Co., Ltd. and China Southern Power Grid Research Institute Co., Ltd. have developed a new renewable energy output prediction method based on the LGBM-Adaboost algorithm. The patent number for this prediction method is CN202410291938.7[P], with a publication date of May 7, 2024.
- [10] Long Huifen. Analysis of User Churn Prediction in Music Streaming [D]. Guangxi: Guangxi Normal University, 2018.
[DOI:10.7666/d.D01508739](https://doi.org/10.7666/d.D01508739).
- [11] Lü Ning Luo Qian: A Telecommunications Customer Churn Prediction Model Integrating XGBoost and Logistic Regression Algorithms. School of Information and Communication Engineering, Beijing University of Information Science and Technology. 10.16652/j.issn.1004-373x.2025.11.021 2025-06-04
- [12] Yan Chun, Zhang Xinyu: A Study on Life Insurance Customer Churn Prediction Algorithms Based on Improved K-means and BP-Adaboost School of Mathematics and Systems Science, Shandong University of Science and Technology 10.16452/j.cnki.sdkjzk.2022.01.006 2022-01-27
- [13] Li Weikang, Yang Xiaobing: A Customer Churn Prediction Model Based on a Two-Layer Fusion Architecture, School of Information Engineering, China Jiliang University, September 9, 2020
- [14] Zhou Jie, Yan Jianfeng, Yang Lu, Xia Peng, Wang Meng: Application of LSTM Model Ensemble Methods in Customer Churn Prediction
- [15] School of Computer Science and Technology, Soochow University, November 29, 2019