

Application of sample balance-based multi-perspective feature ensemble learning for prediction of user purchasing behaviors on mobile wireless network platforms

Huiping Zhang*, Junchao Dong

Guangxi Key Laboratory of Trusted Software, Guilin University of

Electronic Technology, Guilin 541004, China

Email:yirao2002927aor@163.com

Abstract: In recent years, with the rapid development of wireless communication network, M-Commerce has achieved great success. Relying on mobile phones, tablets and other wireless communication devices for online shopping has become a mainstream way for users to consume. Users leave a lot of historical behavior data when shopping on the M-Commerce platform. Using these data to predict future purchasing behaviors of the users will be of great significance for improving user experience and realizing mutual benefit and win-win result between merchant and user. Therefore, a sample balance-based multi-perspective feature ensemble learning was proposed in this study as the solution to predicting user purchasing behaviors, specifically including: 1) “Sliding window”-centroid under-sampling was combined with sample balance method was used, while the positive sample size was enlarged using “sliding window”, centroid under-sampling was used to reduce the negative sample size within “sliding window”, so as to acquire user’s historical purchasing behavioral data with sample balance. 2) Influence feature of user purchasing behaviors were extracted from three perspectives—user, commodity and interaction, in order to further enrich the feature dimensions. Meanwhile, feature selection was carried out using XGBSFS algorithm. 3) An ensemble learning model—five-fold cross validation stacking—which could be used to predict user purchasing behaviors was raised. Three prediction models—XGBoost-Logistics, LightGBM-L2 and cascaded deep forest models—so that they could realize mutual collaboration and the overall prediction ability of the ensemble learning model could be improved. 4) Large-scale real datasets were experimented on Alibaba M-Commerce platform. The experimental results show that the proposed method has achieved better prediction effect in various evaluation indexes such as precision and recall rate.

Keywords: wireless communication network; M-Commerce; ensemble learning; XGBoost-Logistics; LightGBM-L2; cascaded deep forest

I. Introduction

As a new mode of E-Commerce, M-Commerce makes use of the advantages of mobile wireless network and is a beneficial supplement to traditional E-Commerce. M-Commerce is the E-Commerce that uses smart phones, tablets and other wireless terminals for business activities. The perfect combination of the Internet, short distance communication, mobile communication and other information processing technology, so that people can do all kinds of commercial activities without time and space restrictions. In recent years, with the in-depth promotion of M-Commerce, online shopping has gradually become a mainstream consumption mode by virtue of various types, low price and convenient price comparison. At the same time, information overload problem occurs to M-Commerce platforms frequently due to sharp increase of user scale and commodity types. With mass commodity information, the time cost for users to purchase

commodities is increased and market competition of merchants becomes more intense. Therefore, accurately predicting user purchasing behaviors and automatically recommending commodities which may be purchased by users will be of great significance for users to rapidly select and purchase commodities, merchants to carry out precision marketing, and M-Commerce platforms to improve service quality. Using mass data information generated by users when shopping on M-Commerce platforms in the past, digging user preferences and understanding user demands is an effective path to realize prediction of user purchasing behaviors.

The existing prediction models of user purchasing behaviors are mainly divided into collaborative filtering (CF) model and classification model. In the beginning, collaborative filtering, one of the most classical algorithms in recommendation system, was extensively applied to prediction of user purchasing behaviors ^[1]. With gradual expansion of buyer scale on M-Commerce platforms in recent years, the scale of historical behavioral data is also increasing sharply. Data sparsity and low accuracy problems of CF algorithm have become increasingly prominent ^[2]. In addition, CF algorithm excessively relies upon user ratings, so it cannot accurately predict user purchasing behaviors without user ratings or with user rating errors. Therefore, numerous researchers have proposed individual learning prediction models ^[3,4] such as logistic regression, support vector machine (SVM), multilayer perceptron (MLP) and neural network as well as ensemble learning prediction models ^[4-6] such as gradient boosting decision tree (GBDT) and XGBoost. In these models, the prediction of user purchasing behaviors is regarded as a binary classification problem in machine learning (positive example: purchase, negative example: not purchase). After features are established according to user's historical behavioral data, individual learning model or ensemble learning model will be trained to classify user purchasing behaviors. These models are more applicable to the prediction task of user purchasing behaviors based on large-scale data than CF algorithm, where ensemble learning prediction models are generally superior to individual learning prediction models ^[3]. However, most of the existing ensemble learning prediction models are based on traditional ensemble algorithms with weaker representation learning ability than neural network. Furthermore, the existing ensemble learning models are accompanied by problems of single perspective and few dimensions in feature construction. Ensemble learning has achieved considerable development in recent years, new-type ensemble algorithms like LightGBM have displayed their outstanding performance in various fields ^[7], but they are seldom seen in the prediction of user purchasing behaviors.

From the angle of big data analysis and based on historical behavioral data of users on Alibaba M-Commerce platform in the past one month (2014.11.18-2014.12.18), multi-dimensional influence features of user purchasing behaviors were extracted from three perspectives: user, commodity and interaction. XGBoost-Logistics (XL), LightGBM-L2 (LL) and cascaded deep forest (CDC) user purchasing behavioral prediction models were constructed. In the end, FCV-Stacking (FCVS) method was used to integrate the three prediction models and form the final ensemble learning prediction model, thus realizing the prediction of user's future (2014.12.19) purchasing behaviors.

The main contributions of this paper are summarized as follows.

(1) A FCVS ensemble learning-based prediction model for user purchasing behaviors was proposed. This ensemble learning model organically combined XL, LL and CDC prediction models, which considerably improved the prediction accuracy of the ensemble learning method.

Three prediction models—XL, LL and CDC—were put forward, where XL and LL had stronger ability to prevent overfitting than original model, and CDC model solved the sparse connectivity problem existing in original deep foreign and strengthened the model stability.

(2) A “sliding window”-centroid under-sampling combined sample balance method was proposed. While positive sample size was enlarged in “sliding window”, the negative sample size in “sliding window” was reduced through centroid under-sampling. This effectively solved the sample unbalance problem of user’s historical purchasing behaviors and could improve the performance of the user purchasing behavior prediction model.

(3) A multi-perspective extraction method for influence features of user purchasing behaviors was presented. A total of 204 features were extracted from three perspectives: user, commodity and interaction, and feature dimension was about 36 times of original dataset. The constructed features could be migrated to other problems in the M-Commerce field, e.g. prediction of user repeated purchasing behaviors and prediction of user click-through rate.

(4) The experiment was implemented on dataset of Alibaba M-Commerce platform extensively used in this field to verify the effectiveness and superiority of FCV-Stacking ensemble learning model in the prediction of user purchasing behaviors.

The remainder of this paper is organized as follows: Section two expounds research work regarding user purchasing behavior prediction models and ensemble learning; Section three introduces data preprocessing work, mainly including problem description, data cleaning and sample balance; Section four is about feature, including feature construction, feature processing and feature selection; Section five describes three prediction models—XL, LL and CDC—as well as FCVS ensemble learning-based user purchasing behavioral prediction model in details; Section six gives experimental results and conducts discussion; In the end, Section seven summarizes the whole paper and prospects the future research work.

II. Related work

2.1 Collaborative filtering algorithm

Since being proposed in the 1990s, CF algorithm has been widespread in the industrial circles and academic circles and attracted attention from numerous scholars ^[8]. According to algorithm principles, CF algorithm can be mainly divided into two types: memory-based CF and model-based CF ^[1,2], where the former seeks for neighbors having high similarity to user or commodity mainly through user’s historical information and predicts user purchasing behaviors according to their comprehensive commodity evaluation. Memory-based CF has data sparsity problem with weak extensibility ^[9]. Different from memory-based CF, model-based CF trains model mainly through user ratings for commodities and then predicts user purchasing probability ^[10]. At present many model-based CF algorithms have been put forward, where matrix decomposition has gradually become the mainstream method in model-based CF algorithms by virtue of transformation of high-dimensional sparse user rating data and excellent extensibility ^[8]. The main idea is to decompose the original user-commodity rating matrix into low-rank potential matrix through technologies like singular value decomposition (SVD) and then obtain the prediction result through an analysis ^[12]. Matrix decomposition method has relieved the sparsity problem of CF algorithm to a certain degree, but cold start problem remains to be solved, so it is impossible to predict the probability for user to purchase new commodities with difficult similarity calculation ^[12]. Furthermore, CF algorithms transform the prediction problem of user purchasing behaviors into processing of rating prediction problem, and the prediction result highly

depends on user rating information for commodities. A large number of the existing researches have found that falsity and arbitrariness problems exist in user rating information for commodities [13], which restricts the prediction accuracy of CF algorithms.

2.2 Individual learning prediction model

Individual learning prediction model constructs the features of influence factors and inputs into single machine learning or deep learning algorithms for predicting user purchasing behaviors. Directing high cost and low prediction accuracy problems in prediction of user purchasing behaviors, Literature [14] constructed a logistic regression model under two scenarios—promotion day and non-promotion day—after multi-angle analysis in order to explore into influence features of user purchasing behaviors under different scenarios. Literature [3] constructed three feature groups—browse, purchase and collect-add to cart—according to user’s historical activity record and proposed a prediction framework for user purchased brands using multiple machine learning methods like linear regression, Naïve Bayes and SVM as well as threshold moving method. Tang et al. put forward a purchasing behavior prediction framework through Firefly Algorithm-based SVM model with optimized parameters, and obtained an effect superior to SVM model [15]. Sakar et al. compared three prediction models for user purchasing intention—random forest (RF), SVM and MLP—through experiments and found that the accuracy and F1 score of MLP were evidently higher than those of RF and SVM [16]. However, the above models have disadvantages of weak feature representation ability and not high accuracy when used to process user’s historical behavioral data which are quite complicated. Therefore, CNN and RNN-represented deep learning-based prediction models for user purchasing behaviors have been put forward in succession [4,17]. Song et al. used user’s historical purchasing behavioral data, predicted buyer purchasing time based on MLP and RNN models, respectively, and the results showed that MLP achieved better effect than RNN [18]. Ling et al. used full-connected long and short-term memory (FC-LSTM) network to construct an interaction model between client and promotion channel as well as nonlinear serial correlation and accumulation effect between client browsing behaviors, and predicted client purchasing behaviors under multi-channel online promotion [19]. Neural network-based prediction model can greatly strengthen representation ability of mass historical behavioral data, but deep learning model is of insufficient interpretability and fails to visualize influence features of user purchasing behaviors, and moreover, it is inconvenient for merchants and M-Commerce platforms to formulate marketing strategies.

2.3 Ensemble learning prediction model

With the development of ensemble learning technology, more and more scholars have constructed ensemble learning prediction models by integrating different individual learning prediction models, in order to improve the accuracy and robustness of prediction results [20]. Common ensemble learning prediction models mainly include GDBT [4-6] and XGBoost [21]. Literature [5] put forward an online-to-offline (O2O) prediction scheme for morrow commodity purchasing behaviors based on mass user behavior logs. The whole solution mainly included feature engineering and ensemble learning. Besides basic features in the feature engineering part, peculiar features of O2O scenario were extracted, and it was proved in practice that peculiar features extracted from practical scenarios would be of great advantages in improving model performance. In the ensemble learning model part, Bagging fusion strategy was used to construct RF and GDBT-based ensemble learning models, and the results showed that GDBT ensemble learning model had higher F1 score than RF ensemble learning model in the O2O prediction of

morrow purchasing behaviors. Literature [6] also used ensemble learning model of GDBT-based learner, but different from fusion strategy in Literature [5], it used Blending fusion strategy with better effect than Bagging fusion strategy. Li et al. adopted Stacking fusion strategy to construct an ensemble learning model of GDBT-based learner and achieved better effect than Bagging and Blending [4]. Literatures [4-6] demonstrated the feasibility of the improved fusion strategy to improve the performance of ensemble learning prediction model, thus providing a theoretical foundation for FCV-Stacking fusion strategy. Zhou et al. put forward a two-layer Multi-Model Stacking Ensemble (MMSE) Learning, where the first layer trained four ensemble algorithms—RF, AdaBoost, GDBT and XGBoost—as base learners, the second layer used XGBoost algorithm to combine the four base learners and output the final prediction result, and the result indicated that its performance was more outstanding than single ensemble algorithm-based prediction model [20]. Hou et al. proved that [21] simple linear model (e.g. logistic regression) was used to replace nonlinear tree model (e.g. XGBoost) in meta learner of Stacking ensemble learning model, and this could improve the accuracy and generalization ability of ensemble learning model [22]. Therefore, logistic regression model is applied to meta learner in FCV-Stacking fusion strategy.

III. Methods

3.1 Data source and problem description

Data used in this study came from complete behavioral data of twenty thousand users and million-class commodity information provided by Alibaba “Tianchi Big Data Competition” [23]. The data included two parts: user behavioral data (D) on the complete commodity set as shown in Tab. 1; information data (P) on commodity subset as shown in Tab. 2.

Tab. 1 User behavioral data (D)

Field	Field description	Extraction explanation
user_id	User ID	Sampling & field desensitization
item_id	Commodity ID	Field desensitization
behavior_type	User behavioral types for commodities	Including browse, collect, add to cart and purchase, corresponding values are 1, 2, 3 and 4
user_geohash	Space ID of user location, it can be null	Generated through secure algorithm based on longitude and latitude
item_category	Commodity classification ID	Field desensitization
time	Behavior time	Accurate to hour

Tab. 2 Commodity information data (P)

Field	Field description	Extraction explanation
item_id	Commodity ID	Sampling & field desensitization
item_geohash	Space ID of commodity location, it can be null	Generated through secure algorithm based on longitude and latitude
item_category	Commodity classification ID	Field desensitization

The main problem to be solved in this paper is, according to historical behavioral data (D) within one month (November 18—December 18, 2014), to predict user purchasing behaviors on the designated commodity subset (P) on December 19. The prediction results of the example are listed in Tab. 3.

Tab. 3 Example of prediction results

User_id	Item_id
100000	2345
100000	2478
100001	127900
100002	207245

3.2 Data cleaning

1) Processing of missing values

By observing the datasets, it was found that data formats were well structured without messy codes. No missing values were found in other fields except for user_geohash. The percentage of missing values in user_geohash field reached over 2/3, and this field consisted of longitude and latitude generated through a secure algorithm. If missing values were processed using sample padding method, data “distortion” could be easily caused, and it would be difficult for padding data match with the existing enciphered data, so data in this column were directly excluded. Even though influence features of positional information of user purchasing behaviors were missing, data authenticity was guaranteed, and “data skew” problem in model training process was avoided.

2) Processing of abnormal values

After missing values were processed, user purchasing behavioral records of data concentration were further analyzed, and two types of abnormal users, namely users with purchasing record being 0 and those with browse quantity being far greater than purchase quantity, were found. It was speculated the two types of users were “crawler users” who would lead to severe “skew” of user features and be to the disadvantage of establishment of purchasing behavioral prediction model. Hence, the following four rules were set, and data of one user would be excluded only if one of the rules was satisfied:

(1) Never collected items or added to cart or purchased them from November 18 to December 18, 2014;

(2) Browse quantity was larger than 400 but had no behaviors of purchase, collect or add to cart from November 18 to December 18, 2014;

(3) Browse quantity was larger than 1,000 without purchasing behavior, but added to cart or collected from November 18 to December 18, 2014;

(4) Browse quantity was 4,000 and $\frac{\text{Browse}}{\text{Purchase}} > 400$ from November 18 to December 18 in 2014.

According to the above rules, a total of 310,104 data were excluded, accounting for about 2.03% of the original data size. The user quantity was 508, accounting for 2.54% of the original data size. The commodity quantity was 69,757, accounting for 2.42% of the original data size. The quantity of commodity categories was 32, account for 3.57% of the original data size. It could be known that the excluded data size occupied a small proportion in the whole dataset, and following the exclusion, the data were still concentrated and mass data information were reserved.

3.3 Sample construction and balance

The prediction of user purchasing behaviors can be solved as a binary classification problem in machine learning. First, it is necessary to construct data samples and datasets. Due to small

positive sample size and severe unbalance between positive and negative samples (1:45) in historical behavioral data in the dataset, the already known data one month before (November 18—December 18) the behavioral prediction date (December 19) could not be simply extracted as data samples, which formed the training set of the user purchasing behavioral prediction model. Given this, a “sliding window”-centroid under-sampling combined balance method was raised to construct training set and test set [5,24].

User’s historical behaviors were divided into: browse, collect, add to cart and purchase. Daily average sales volume and the scale of users who purchased commodities in the existing data samples were calculated to investigate the overall fluctuation tendency of commodity sale as shown in Fig. 1.

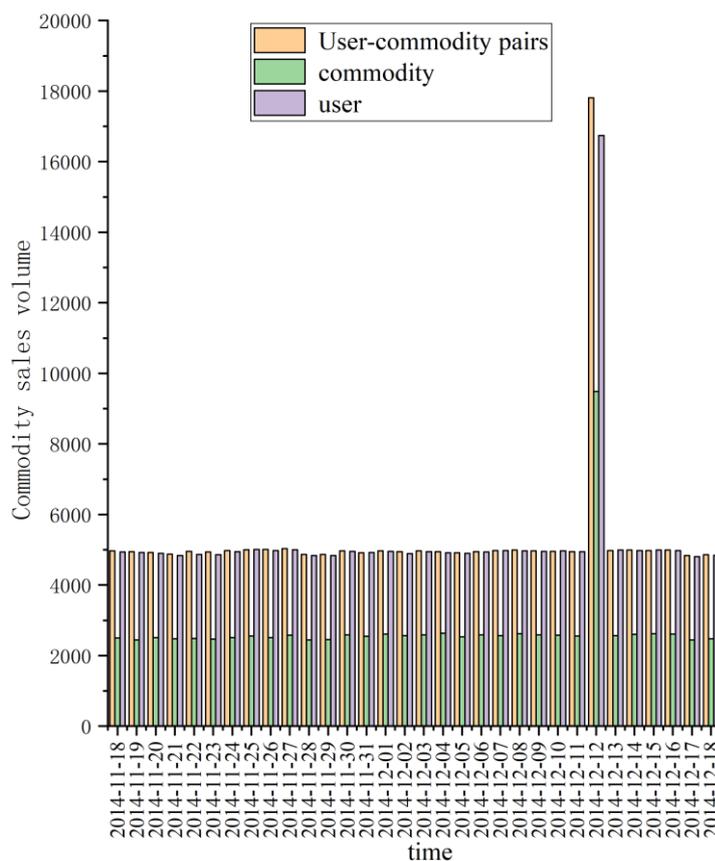


Fig. 1 Commodity purchase conditions every day within one month

From Fig. 1, because discount promotion activity was launched on Alibaba on December 12, the user activeness was high, especially commodity sales volume broke through 18,000, exceeding three times of daily sales volume on other dates. Except for December 12, daily sales volume on other dates fluctuated at 5,000, and the fluctuation range was small. Severe sample deviation in the training set would be caused due to abrupt growth of data size on December 12. To standardize extreme deviations of behavioral logs, smooth actors were added to numbers of times of four behaviors—browse, collect, add to cart and purchase—on “Double Twelve” (December 12) [5], specifically being 0.50, 0.67, 0.43 and 0.22, respectively.

The size of “time window” depended upon the time span of user purchasing behaviors and

other historical behaviors (browse, collect and add to cart). A specific day taken as benchmark time point, user-commodity ID pairs of all purchasing records on this day were acquired, the records of four historical behaviors—browse, collect, add to cart and purchase—within a time period before the benchmark time point were organized, and the time-dependent change trajectories of the occurrence frequency of the four interactive behaviors were respectively obtained. The experimental results when November 1 was taken as the benchmark time point were selected as shown in Fig. 2.

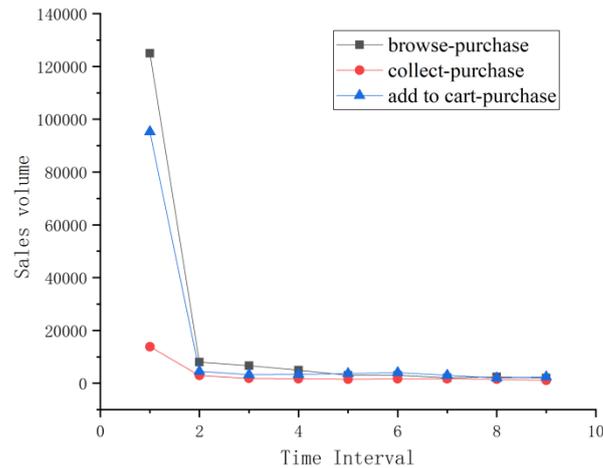
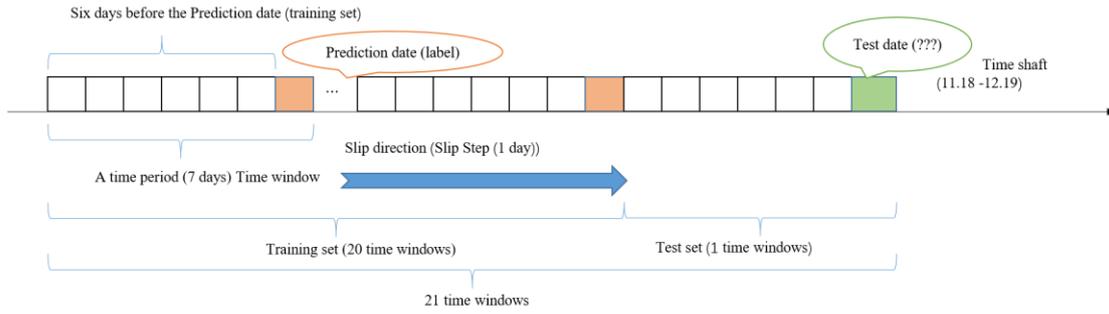


Fig. 2 Time difference-dependent quantity changes of three behaviors and purchasing behavior

As shown in Fig. 2, the record quantities of browse, collect and add to cart all evidently declined with the purchasing date forward, the decrease amplitude was gradually slowed down, and it gradually tended to be steady 7 days later. It could be known that user purchasing behaviors were closely related to other behaviors within the 7 days, so “time window” size was set as 7 days.

After the 7-day “time window” was determined, “sliding window” method was used to construct training set and test set. The original data included 31 days, the 7-day time window slid forward for one day each time from the benchmark date (November 1), and a training set including 20 time windows and a test set including one time window were finally constructed. In each time window of the training set, the first 6 days were trained as features and the 7th day served as label data. If a user purchased one commodity on the 7th day, it would be labeled as positive sample, or otherwise negative sample. The positive sample size was expanded to about 20 times of the original sample size through the “sliding window” method. In addition, user behavioral data within the first 6 days (2014.12.13-2014.12.18) before the prediction date (December 19) were taken as the test set. The concrete process is displayed in Fig. 3.



Slide number	Training set						label
	1	11.18	11.19	11.20	11.21	11.22	11.23
2	11.19	11.20	11.21	11.22	11.23	11.24	11.25
3	11.20	11.21	11.22	11.23	11.24	11.25	11.26
4	11.21	11.22	11.23	11.24	11.25	11.26	11.27
5	11.22	11.23	11.24	11.25	11.26	11.27	11.28
6	11.23	11.24	11.25	11.26	11.27	11.28	11.29
7	11.24	11.25	11.26	11.27	11.28	11.29	11.30
8	11.25	11.26	11.27	11.28	11.29	11.30	11.31
9	11.26	11.27	11.28	11.29	11.30	11.31	12.1
10	11.27	11.28	11.29	11.30	11.31	12.1	12.2
11	11.28	11.29	11.30	11.31	12.1	12.2	12.3
12	11.29	11.30	11.31	12.1	12.2	12.3	12.4
13	11.30	11.31	12.1	12.2	12.3	12.4	12.5
14	11.31	12.1	12.2	12.3	12.4	12.5	12.6
15	12.1	12.2	12.3	12.4	12.5	12.6	12.7
16	12.2	12.3	12.4	12.5	12.6	12.7	12.8
17	12.3	12.4	12.5	12.6	12.7	12.8	12.9
18	12.4	12.5	12.6	12.7	12.8	12.9	12.10
19	12.5	12.6	12.7	12.8	12.9	12.10	12.11
20	12.6	12.7	12.8	12.9	12.10	12.11	12.12
Test set							result
12.13	12.14	12.15	12.16	12.17	12.18	12.19	

Fig. 3 Construction process of training set and test set through “sliding window” method

The positive sample size was enlarged by about 20 times through the “sliding window” method, which relieved the user purchasing behavioral sample unbalance problem in the dataset to a certain degree. To further solve the sample unbalance problem, while the positive sample size was enlarged in “sliding window”, centroid under-sampling (ICIKMDS) was adopted to perform under-sampling of negative samples (large-class samples) in “time window”. With a view to sampling technology level, the samples in large class were under-sampled through this method, which could guarantee the information content in large-class samples to the greatest extent while reducing the large-class sample size. First, the initial centroid in large-class samples was found, followed by clustering using K-means algorithm, and k clusters were acquired. The sample in each cluster, which had the maximum similarity with the centroid of this cluster, was selected, thus finally forming the final set of large-class samples. The concrete algorithm description is shown in Tab. 4.

Tab. 4 ICIKMDS algorithm

Input: Dataset $S = \{(x_1, y_1), (x_1, y_1), L, (x_{n+m}, y_{n+m})\} = X^+ \cup X^-$, where small-class positive sample set is $X^+ = \{(x_1^+, y_1^+), L, (x_m^+, y_m^+)\}$ and large-class negative sample set is $X^- = \{(x_1^-, y_1^-), L, (x_n^-, y_n^-)\}$, and $n \gg m$

Output: Balanced dataset S^*

Process:

1 //Acquire $k(k = m)$ initial clustering centers
 $\{c_1, c_2, L, c_m\} = \text{FindInitialcenter}(X^-, m)$
repeat
 2 **for each** $x_i \in X^-, i = 1, 2, L, n$ **do** //traverse each sample in the negative sample set
 3 //Select c_j , which has the maximum similarity with sample x_i , in initial clustering center set $\{c_1, c_2, L, c_m\}$, and make it a member of this data cluster

$$\text{sim}(x_i, c_j) = \frac{x_i \cdot c_j}{\|x_i\| \|c_j\|}$$

4 **end for**
 5 //Update centroid, and obtain the new centroid set $\{c'_1, c'_2, L, c'_m\}$

$$c'_j = \frac{\sum_i x_i}{\text{Number of samples belonging to the present centroid } C_j},$$

where x_i is a sample belonging to centroid C_j

6 **until** the centroids do not undergo any change or reach the convergence requirement
 7 Acquire the final centroid set $\{c_{1\text{-final}}, c_{2\text{-final}}, L, c_{m\text{-final}}\}$
 8 //Select the sample in each cluster, which has the maximum similarity with the centroid of this cluster
 $x^{-i} = \arg \max_{x_i \in C_{i\text{-final}}} \text{sim}(x_i, c_{i\text{-final}})$
 9 //Obtain m representative samples to constitute the final large-class sample sets

 x^- and x^+ **return** S^*

4. Experiment

Data scale and feature engineering quality decide the upper limit of the subsequent prediction model to a certain degree. Therefore, feature construction, processing and selection were carried out before the modeling of user purchasing behavioral prediction.

4.1 Feature construction

User's historical purchasing behavioral data after data preprocessing in Section three included five usable fields: user ID, commodity ID, user behavioral type, commodity type and user behavior time, which could be directly regarded as behavioral features with too few dimensions, failing to accurately express the laws of user's historical purchasing behaviors, so the training effect would be extremely poor if they were directly input into the mode. Hence, based on a summary of feature engineering in the existing researches as discussed in Section two, the business logic of the M-Commerce field was combined to reconstruct and fuse the information of the existing five fields from three perspectives: user, commodity and user-commodity interaction. The hidden information contained in the original data were deeply dug to enrich the feature dimensions of influence factors of user purchasing behaviors, expecting to improve the accuracy of the prediction model.

1) User perspective

User purchasing habit and preference are of vital importance to prediction of user purchasing behaviors. From five effective fields in the original data, 26 user perspective features as shown in Tab. 5 were constructed from three aspects—statistics, proportion and time—through methods like statistical analysis based on two commodity types ($i=\{1: \text{commodity}, 2: \text{commodity classification}\}$), four user behavioral types ($d=\{1: \text{browse}, 2: \text{collect}, 3: \text{add to cart}, 4: \text{purchase}\}$) and three time granularities from the prediction date ($n=\{1,3,6\}$).

Tab. 5 Detailed description of user perspective features

Feature type	Symbol	Definition	Meaning	Quantity
Statistical feature	u_n_counts	User behavioral quantity n days before the prediction date	Reflect user activeness in different time periods	3
	u_n_d_counts	The quantity of d type user behaviors n days before the prediction date	Reflect the activeness of d type user behaviors in different time periods	12
Proportional feature	u_n_b_rate	Ratio of user purchasing behavior to other three behaviors n days before the prediction date	Reflect the conversion ratio of user purchasing	9

			behaviors in different time periods	
	u_active_time	Ratio of number of days with user behavior records to total number of days	Reflect user enthusiasm in the whole time period	1
Time feature	u_b_time	Time difference between user browsing behavior and purchasing behavior	Reflect time spent in user behavior decision making	1

2) Commodity perspective

The heat degree of commodities or commodity classification was mainly analyzed from the commodity perspective. Besides three aspects same as user perspective, the heat degree ranking feature of commodities in the classes where they were located was newly increased. A total of 76 features were constructed in the end from commodity perspective as seen in Tab. 6.

Tab. 6 Detailed description of commodity perspective features

Feature type	Symbol	Definition	Meaning	Quantity
Statistical feature	i_n_ucounts/ c_n_ucounts	User quantity for commodity/commodity classification n days before the prediction date	Reflect popularity of commodity/commodity classification among users	6
	i_n_d_bcounts/ c_n_d_bcounts	Quantity of type d behaviors for commodity/commodity classification n days before the prediction date		24
	i_n_d_bucounts/ c_n_d_bucounts	Quantity of users having type d behaviors for commodity/commodity classification n days before the prediction date		24
Proportional feature	i_n_b_rate/ c_n_b_rate	Ratio of purchasing behavior quantity to the quantity of the other three behaviors for commodity/commodity classification n days before the prediction date	Reflect purchasing conversion ratio of commodity/commodity classification	18
Time feature	i_b_time/	Average time difference	Reflect average time	2

	c_b_time	between browsing behavior and purchasing behavior for commodity/commodity classification	spent in purchasing decision making for commodity/commodity classification	
Ranking feature	ic_rank	Commodity ranking in behavioral quantity in the classification it belongs to	Reflect the heat degree of commodity in the classification it belongs to	1
	ic_b_rank	Ranking of purchasing behavior quantity for commodity in the classification it belongs to		1

3) Interactive perspective

As a combinational perspective of user perspective and commodity perspective, interactive perspective reflected the behavioral relation between user and commodity. The four aspects similar to commodity perspective were used to construct 102 features (Tab. 7).

Tab. 7 Detailed description of interactive perspective features

Feature type	Symbol	Definition	Meaning	Quantity
Statistical feature	u_i_n_counts/ u_c_n_counts	Behavior quantity of one user for one commodity/commodity classification n days before the prediction date	Reflect enthusiasm of one user for one commodity/commodity classification	6
	u_i_n_d_bcounts/ u_c_n_d_bcounts	Quantity of type d behaviors of one user for one commodity/commodity classification n days before the prediction date		24
Proportional feature	u_i_n_d_rate/ u_c_n_d_rate	Ratio of purchasing behavior quantity to the quantity of the other three behaviors of one user for one commodity/commodity classification n days before the prediction date	Reflect purchasing conversion ratio of one user for one commodity/commodity classification	18
	u_i_n_b_rate/	Ratio of purchasing		Reflect preference of

	u_c_n_b_rate	behavioral quantity of one user for one commodity/commodity classification n days before the prediction date to total quantity of the commodity/commodity classification purchased n days before the prediction date	one user for one commodity/commodity classification	
	u_i_n_u_rate/ u_c_n_u_rate	Ratio of quantity of type d behaviors of one user for one commodity/commodity classification n days before the prediction date to total quantity of type d behaviors of the user n days before the prediction date		24
Time feature	u_i_b_time/ u_c_b_time	Time period from the first type d behavior of one user for one commodity/commodity classification until the last time	Reflect time span for repeated behavior of one user for one commodity/commodity classification	8
	u_i_b_p_time/ u_c_b_p_time	Time period of first type d behavior of one user for one commodity/commodity classification until the prediction date	Reflect loyalty of one user to one commodity/commodity classification	8
Ranking feature	u_ic_b_rank	Ranking of quantity of type d behaviors of one user for one commodity/commodity classification in the total quantity of type d behaviors of the user for all commodities/commodity classifications	Reflect preference of one user for one commodity/commodity classification	8

The 204 features constructed from the three perspectives were combined according to user

number and commodity number to form a new data sheet, and meanwhile, in accordance with the requirements of “sliding window” specified in section 3.3, the purchasing behaviors of this user on the seventh day within the time window were labelled, and labelling results were added after multi-perspective feature field in the data sheet as a new column. The labeling rule was: it would be labeled as 1 if the commodity was purchased, or otherwise as 0. The final data format is shown in Fig. 4.

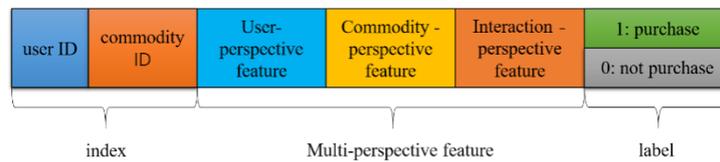


Fig. 4 Data format of each row in the new data sheet

4.2 Feature processing

The features constructed based on user’s historical purchasing behavioral data from three perspectives in section 4.1 were not unified in dimension and unit, which would influence the model to evaluate feature weights and then influence its accuracy and convergence rate. Therefore, Min-Max standardization was adopted for feature normalization, and feature data were zoomed into [0,1] interval. The transfer function is as below:

$$X^* = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \quad (1)$$

Where Max is maximum value of sampled data; Min is minimum value of sampled data.

4.3 Feature selection

As there were numerous influence factors of user purchasing behaviors, the historical behavioral data structure was complex, the number of constructed features was too large, the training time cost would be too high if they were directly input into the prediction model established in Section five, and moreover, the model complexity could be aggravated due to the multicollinearity problem between features, overfitting problem could be easily triggered, and then model robustness would be degraded [25]. Therefore, XGBSFS feature selection method was used to select feature subsets with important effects on purchasing behavioral prediction from many datasets, so as to improve the model prediction accuracy and shorten operating time [26].

XGBSFS is an XGBoost-based wrapper-type feature selection method. The tree construction process of XGBoost algorithm was used, feature importance was measured using F-Score, Average Gain and Average Cover, respectively, followed by searching through an improved sequential floating forward selection (ISFFS) strategy, and in the end, the feature subset with the highest classification accuracy was taken as the feature selection result. The concrete steps are seen in Tab. 8.

Tab. 8 XGBSFS feature selection algorithm

Input:	Original feature set A
Output:	Target feature set O
Process:	
1	$O \leftarrow \emptyset, \text{depth} \leftarrow \emptyset$
2	Initialize importance measure

```

3   Build root node
4   while  $depth < \text{maximum tree depth}$  do
5       for each  $a \in A$  do
6           Calculate the optimal segmentation point  $a^k$ 
7           Build child node for the optimal segmentation point  $a^k$ 
8       end for
9        $depth \leftarrow depth + 1$ 
10  end while
11  //Add the built tree into tree model
12   $\mathcal{Y}_i^{(t)} = \sum_{k=1}^K f_k(x_i) = \mathcal{Y}_i^{(t-1)} + f_t(x_i), f_k \in F$ 
13  for each tree do
14      for each segmentation node do
15          Calculate gain value generated by segmentation node
16          Calculate sample number (cover) within the three
17          // Update importance measure of each feature
18          F-Score  $\leftarrow$  F-Score + 1, Gain  $\leftarrow$  Gain + gain, Cover  $\leftarrow$  Cover + cover
19      end for
20  end for
21  for each  $a \in A$  do
22       $AverageGain \leftarrow \frac{Gain}{FScore}, AverageCover \leftarrow \frac{Cover}{FScore}$ 
23  end for
24  Sort features according to importance measure
25  Select two candidate feature subsets  $I_1$  and  $I_2$ , where  $I_1$  is a set of features sorted
    in a descending order according to important measure  $i_1$  and  $I_2$  is a set of features
    sorted in an ascending order according to importance measure  $i_2$ 
26  Acquire feature from the set  $I_1$  so that the accuracy evaluation function  $J$  is
    increased after  $x_b$  is added into the target set  $O$ 
27  while  $x_b$  exists do
28       $O \leftarrow O \cup x_b$ 

```

```

27   Acquire feature from  $I_2$  so that  $J$  is increased after  $O$  deletes  $x_w$ .
28   while  $x_w$  exists do
29        $O \leftarrow O - x_w$ 
30   repeat acquire  $x_w$  from  $I_2$ 
31   end while
32   repeat acquire feature  $x_b$  from  $I_1$ 
33   end while
34   return  $O$ 

```

5. Results

5.1 Model framework

Fig. 5 shows the ensemble learning model framework for predicting user purchasing behaviors, which integrates user, commodity, commodity classification and user-commodity interaction information. First, data cleaning of user's historical behaviors was completed, and the samples were balanced using the "sliding window"-ICIKMDS method. User perspective features, commodity perspective features and user-commodity interactive perspective features were constructed from three perspectives: user, commodity and interaction, and they were input into XGBSFS algorithm for feature selection; The selected features were respectively input into XL, LL and CDC prediction models for user purchasing behavior model training and prediction. In the end, the three prediction models were integrated through the FCVS method and the final prediction results were output after classification.

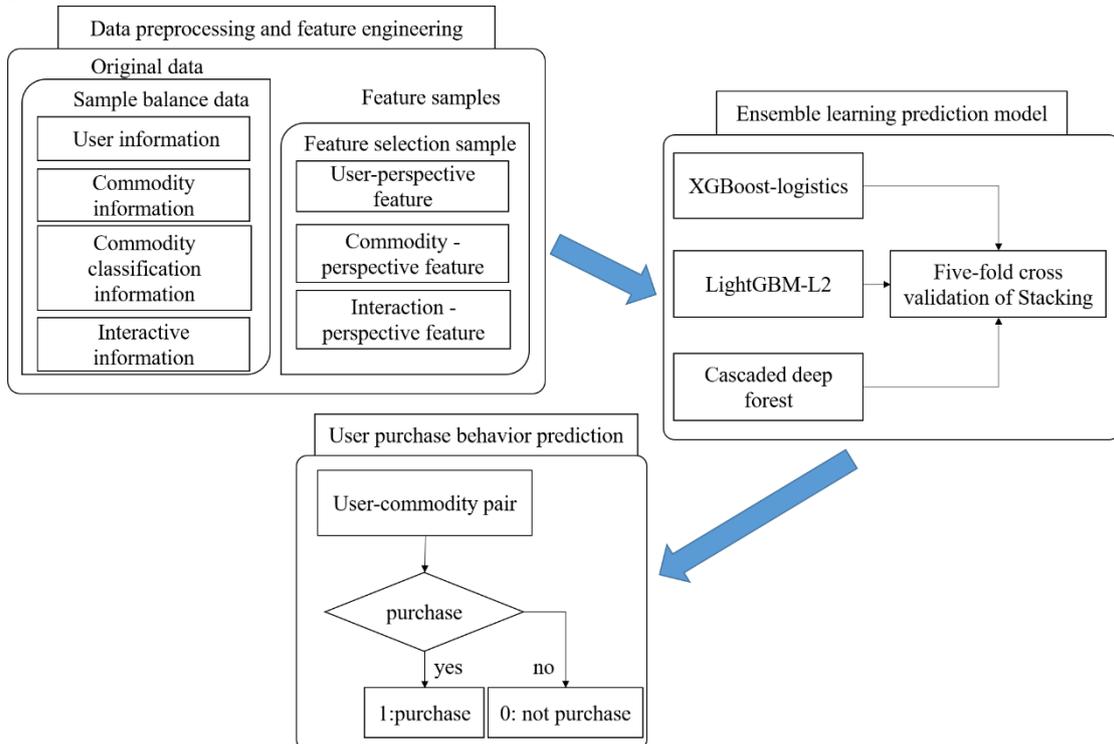


Fig. 5 Framework of user purchasing behavioral prediction model

5.2 XGBoost-Logistics user purchasing behavioral prediction model

5.2.1 XGBoost-Logistics algorithm

XGBoost algorithm was used to predict user purchasing behaviors, mainly including two steps: (1) construct XGBoost algorithm model and use the training set to train the corresponding parameters; (2) Apply the trained model to the test set and calculate the probability for user to purchase commodities.

CART tree is used as base learner in XGBoost algorithm [27]. The objective function of XGBoost user purchasing behavioral prediction model is defined as below:

$$J(\theta) = L(\theta) + \Omega(\theta) \quad (2)$$

Where θ is parameter in many formulas; $L(\theta)$ is loss function used to measure the degree of fitting between model and training data. Commonly used loss functions include quadratic loss and Logistic loss, etc. $\Omega(\theta)$ is a regularizer used to control model complexity. The aim of introducing this regularizer is to ensure that the model generated through the data in the training set can accurately predict the test samples. Not only the model simplicity is considered but also the model training error can be minimized. Through regularization, the model can not only fit the training set very well but also has excellent performance on the test set. Common regularizers are L1 regularizer and L2 regularizer.

User purchasing behavioral prediction belongs to a classification problem, so Logistic classifier is used to express the loss function:

$$L(\theta) = y_i \ln(1 + e^{-\hat{y}}) + (1 - y_i) \ln(1 + e^{\hat{y}}) \quad (3)$$

Theoretically, the output $y_i = XGBoost(x_i)$ of XGBoost algorithm can be an arbitrary value within the scope of $(-\infty, +\infty)$, so it is not applicable to binary classification problem of user purchasing behavioral prediction. Given this, Logistics function as shown in formula (4) is introduced to transform the model output into the scope of $(0, 1)$.

$$y_i^* = \frac{1}{1 + e^{-y_i}} \quad (4)$$

Where y_i^* is probability output, and threshold value $\alpha=0.5$ is selected to acquire the final prediction result.

$$y_i^{pred} = \begin{cases} 0 & y_i^* \leq \alpha \\ 1 & y_i^* > \alpha \end{cases} \quad (5)$$

Formula (5) can transform the output of XGBoost model into two types: purchase and not purchase, and the output probability y_i^* can reflect the certainty of model prediction. The closer the y_i^* value is to 1, the higher the certainty for the model to classify a sample into “purchase”;

The closer the y_i^* value is to 0, the higher the certainty for the model to classify a sample into “not purchase”.

5.2.2 Construction of XGBoost-Logistics prediction model

The construction of XGBoost-Logistics user purchasing behavior can be implemented through the five following steps:

- (1) Generate a decision tree;
- (2) Calculate first-order derivative g_i and second-order derivative h_i at each training sample point;
- (3) Generate a new decision tree through the greedy algorithm and calculate the corresponding predicted value of each leaf node;
- (4) Add the newly generated decision tree $f_t(x)$ to the model for continuous iteration;
- (5) Input the output y_i of XGBoost algorithm into Logistics function, acquire the final prediction result and transform into two types: purchase and not purchase.

The constructed training set was used to train XGBoost-Logistics model, and the trained model was adopted to predict user purchasing behaviors in the test set.

5.3 LightGBM-L2 user purchasing behavioral prediction model

5.3.1 LightGBM-L2 algorithm

As a decision tree-based gradient boosting framework, LightGBM supports efficient parallel training. The traditional GBDT is optimized mainly using histogram optimization algorithm, Leaf-wise decision tree growth strategy with depth limitation, gradient-based one-side sampling (GOSS) and exceptional feature binding (EFB) algorithm. The improved algorithm has the following advantages: higher training efficiency, low memory usage and higher accuracy, it supported parallel and GPU computation, and moreover, it can satisfy the processing requirement for user historical behavioral data with large data size and high feature dimension [28].

In terms of user purchasing behavioral prediction, the consequence of mistakenly classifying “purchase” samples into “not purchase” samples differed a lot from that of mistakenly classifying not purchased samples into purchased samples, and the cost of the former was evidently higher than that of the latter. As sample diversity would be strengthened due to data generated by multiple behavioral types such as browse and collect accompanying user purchasing behaviors, thus causing model overfitting risk, to reduce the model overfitting risk and enhance model robustness, positive sample quantity was enlarged through “sliding window”, and based on ICIKMDS sample balance strategy, L2 regularizer was introduced to improve the loss function of LightGBM, specifically as follow:

The original loss function in tree m is:

$$L_o = -\frac{1}{N} \sum_{i=1}^N L(y_i, F_{m-1}(x_i; P_{m-1})) \quad (6)$$

Where $F_{m-1}(x_i; P_{m-1})$ denotes the predicted value corresponding to x_i input by the model

containing $m-1$ trees under parameter $P_{m-1} = \{p_1, p_2, \dots, p_{m-1}\}$. $L(y_i, F_{m-1}(x_i; P_{m-1}))$ is

logarithmic loss function of true value y_i and the current model predicted value.

After L2 regularizer is introduced for improvement, the new loss function in tree m will be:

$$L_N = -\frac{1}{N} \sum_{i=1}^N L(y_i, F_{m-1}(x_i; P_{m-1})) + \frac{\lambda}{2} \|\omega\|_2^2 \quad (7)$$

Where λ is regularization coefficient, and coefficient ω_i can be obtained through formula (8):

$$\omega_i = \begin{cases} c & y_i = 1 \\ 1 & y_i = 0 \end{cases} \quad (8)$$

Where $y_i = 1$ is purchased sample, $y_i = 0$ is “not purchase” sample, and c is a constant greater than 1 related to sample proportion.

5.3.2 Establishment of LightGBM-L2 prediction model

The main construction process of LightGBM-L2-based purchasing behavioral prediction model is shown in Tab. 9.

Tab. 9 LightGBM-L2 algorithm

Input:	Data D^* through feature selection based on XGBSFS algorithm
Output:	User purchasing behavior type y^* (1: purchase, 0: not purchase)
Process:	
1	//Initialize quantity m of decision tree-based learners and train weight $\frac{1}{m}$ of example
	Initialize $DT(m)$, $\alpha = \frac{1}{m}$
2	while $i < n$ do //model iteration, $i, 0 \leq i \leq n$ is number of iterations
3	train $f_j(x) = DT(\mathfrak{g})$ // $j, 1 \leq j \leq m$ train decision tree-based learner
4	$\alpha \leftarrow \text{loss}(f(x))$ // determine the weight of the current base learner according to training error
5	$f(x) = \alpha_0 f_0(x) + \alpha_1 f_1(x) + \dots + \alpha_m f_m(x)$ // acquire the final classifier
6	$F(x) = \text{Integrate}(f(x))$ //combine base learners as a strong classifier
7	$y^* = F(D^*)$ // use the strong classifier to predict the final result

5.4 Cascaded deep forest prediction model

5.4.1 Cascaded deep forest

The cascaded deep foreign prediction model is a deep forest model improved specific to prediction task of user purchasing behaviors. With low training cost, deep forest is applicable to large-scale user historical behavioral data. The cascaded deep foreign consists of two parts: cascaded forest structure and multi-granularity scanning strategy [31].

1) Cascaded deep forest

By improving the cascade structure of deep forest, the cascaded deep forest solves the sparse connectivity problem existing in deep forest and improves the stability of prediction results of purchasing behaviors. As shown in Fig. 6, each level in the cascade receives the splicing vector of input feature vector and average class distribution vector output by each level of forest, and outputs the processing result of this level to the next level.

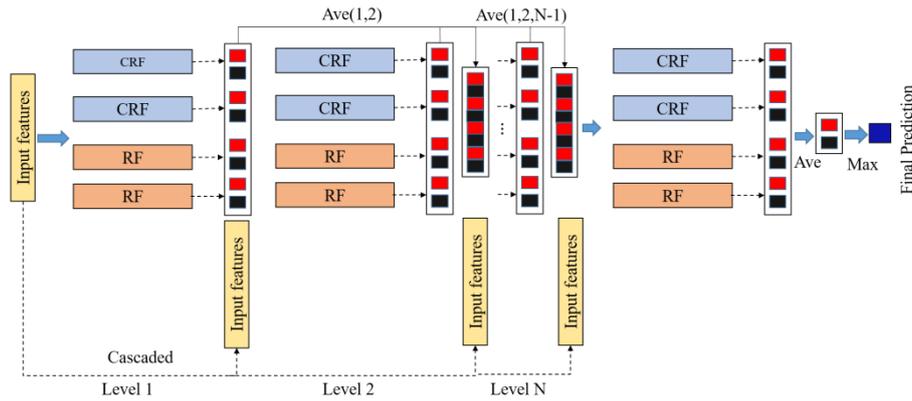


Fig. 6 Structural illustration of cascaded deep forest

To satisfy the basic requirements for great individual model differences in ensemble learning, each level in the cascade structure contains two base learner types: completely-random tree forest and random forest with the same quantity, where the growth rule of the former is: randomly select features in complete feature space and split each tree node until each leaf node only contains the same type of samples or number of samples is not greater than 10. The growth rule of the latter is: construct a candidate feature space containing \sqrt{d} (d is number of input features) random features first and then select features with the minimum Gini coefficient in the candidate feature space for splitting.

Fig. 7 displays the class vector generation process for binary classification problem, namely prediction of user purchasing behaviors. Input sample data x , reach the corresponding leaf node in each tree through a certain decision-making rule, and obtain the proportion of two types of training samples (purchase and not purchase) at leaf node. This node main contains different types of data samples, and then the probability distribution of different types is calculated. The average proportion of leaf node types in all trees in the whole forest is taken and then their probability distribution in the whole forest is obtained. The red line in the figure is the path from sample to leaf node.

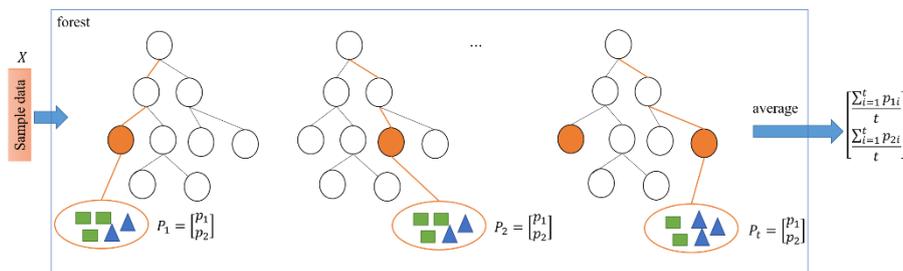


Fig. 7 Class vector generation

2) Multi-granularity scanning strategy

“Sliding windows” of multiple sizes are used by the multi-granularity scanning strategy, the final feature vector after transformation will contain more abundant feature information, thus further improving representation ability of deep forest as shown in Fig. 8.

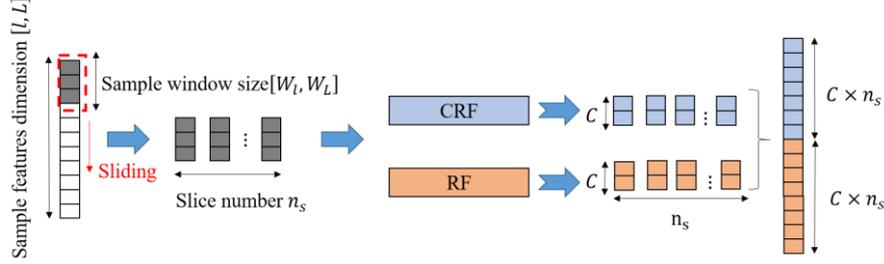


Fig. 8 Scanning strategy of one “sliding window”

Assume total sample size as N , sample dimension as $[l, L]$, label number as C , and then the original size of input data is:

$$I_D = N \times L \times l \quad (9)$$

Slicing input data is the key to realizing multi-granularity deep forest. Assume “sliding window” size as $[W_l, W_L]$ and sliding step size as $[S_l, S_L]$, and then the slice number of each sample is:

$$n_s = \left(\frac{l - W_l}{S_l} + 1\right) \left(\frac{L - W_L}{S_L} + 1\right) \quad (10)$$

Slice size is $W_l \times W_L$, and slice number of the whole dataset is:

$$N_s = N \times W_l \times W_L \left(\frac{l - W_l}{S_l} + 1\right) \left(\frac{L - W_L}{S_L} + 1\right) \quad (11)$$

After multi-granularity scanning, the slices of all input data are input into the random forest to generate class vectors. The number $n_v(w)$ of class vectors of each sample, which are generated through “sliding window” and random forest, is the same as slice number $n_s(w)$ transferred into the random forest:

$$n_v(w) = n_s(w) \quad (12)$$

If there are n_{RF} random forests, the quantity of class vectors generated through “sliding window” sampling is:

$$N_v(W) = N \times n_v(w) \times n_{RF} \times C \quad (13)$$

The final output data size after multi-granularity scanning is:

$$N_{FV}(W) = N \times C \times n_v(w) \sum_w n_{RF} \quad (14)$$

Fig. 9 shows the overall process of cascaded deep forest with three “sliding windows”, the sizes of which are $d/16$, $d/8$ and $d/4$, respectively, where d represents feature number. The data input in the model are original feature samples. Three different feature class vectors are acquired by combining the multi-granularity scanning strategy as shown in Fig. 8. In the end, the class vectors are input into the cascaded deep forest for multilayer vector representation, and the prediction result is output.

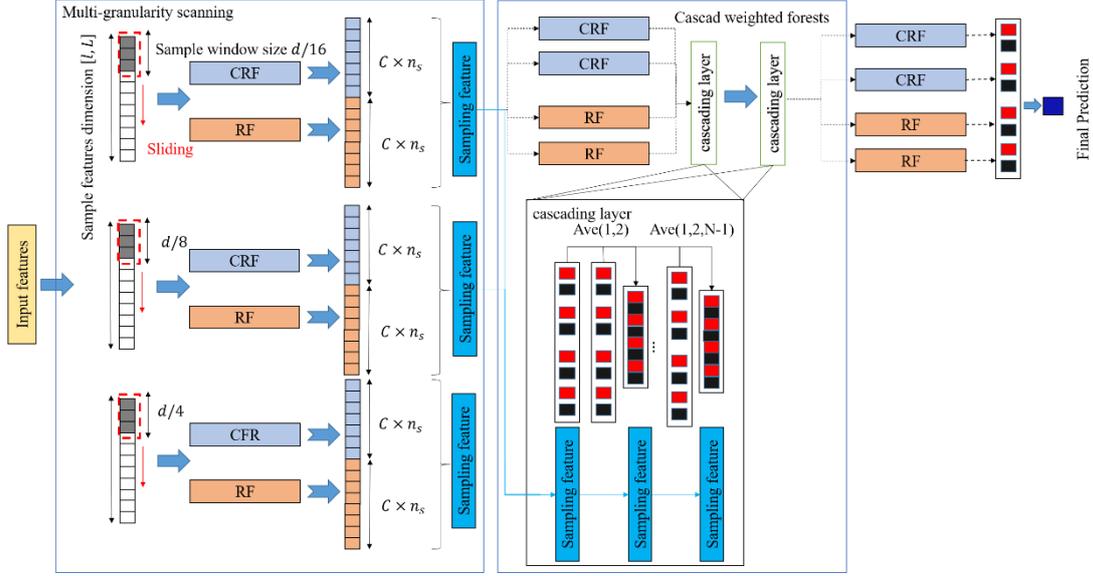


Fig. 9 Cascaded deep forest processing of multi-granularity scanning

5.4.2 Prediction model construction based on cascaded deep forest

The main construction process of purchasing behavioral prediction model based on cascaded deep forest is shown in Tab. 10.

Tab. 10 Cascaded deep forest algorithm

Input: Data D^* through feature selection using XGBSFS algorithm
Output: User purchasing behavior type y^* (1: purchase, 0: not purchase)

Process:

- 1 //Initialize
 $D_0 = D^*$
- 2 //Use data $D_i, i = 1, 2, L, n$ to train two forest forests
 $F_{RF} = \text{tain}(D_i)$
- 3 //Use data $D_i, i = 1, 2, L, n$ to train two completely random forests
 $F_{CRF} = \text{tain}(D_i)$
- 4 //Output a two-dimensional vector X for each forest at layer i

```

 $X_{CRF} = F_{CRF}(D_i), X_{RF} = F_{RF}(D_i)$ 
5 //Solve average value  $\bar{X}$  of two-dimensional class vectors
 $\bar{X} = \text{ave}(X_{CRF}, X_{RF})$ 
6 //Set output type of layer  $i$  as  $\text{argmax}(\bar{X})$ 
 $y = \text{argmax}(\bar{X})$ 
7 Acquire the accuracy  $\theta_i$  of the test result of the current model in the validation set
8 if  $\theta_i < \theta_{i-1}$  then
9     return  $F_{DC}$ 
10 else
11      $X^* = \text{con}(\bar{X}, \bar{X}_{i-1})$  //average value  $\bar{X}_{i-1}$  of two-dimensional class vector
     $X$  output by each forest at layer  $i$  and two-dimensional class vector at the first  $i-1$ 
    layer
13     goto Step (2)
14  $y^* = F_{DC}(D^*)$  //Use cascaded deep forest to predict the final result

```

5.5 Ensemble learning-based user purchasing behavioral prediction model

5.5.1 Five-fold cross validation of Stacking prediction model

Ensemble learning is a machine learning pattern which integrates multiple base learners to solve the same problem^[31]. Generally speaking, the performance of ensemble model exceeds that of base learners so that ensemble learning becomes quite popular^[31-35]. Stacking is a base learner fusion method which has been most extensively applied in the ensemble learning field^[34,35]. However, different base learners at the first layer of the traditional Stacking ensemble model use the same training set, so the differences of their output values are minor, which leads to poor model generalization performance. Therefore, a two-layer Stacking model of five-fold cross validation (FCV-Stacking) is proposed as shown in Fig.10. The first layer consists of three base learners: XL, LL and CDC, which are trained respectively using the five-fold cross validation strategy. The prediction results of base learners are input into the logistic regression meta learner at the second layer to train final user purchasing behavioral prediction model. In the end, output results of test samples obtained through base learners are combined and input into the trained logistic regression meta learner to conduct final prediction of user purchasing behaviors.

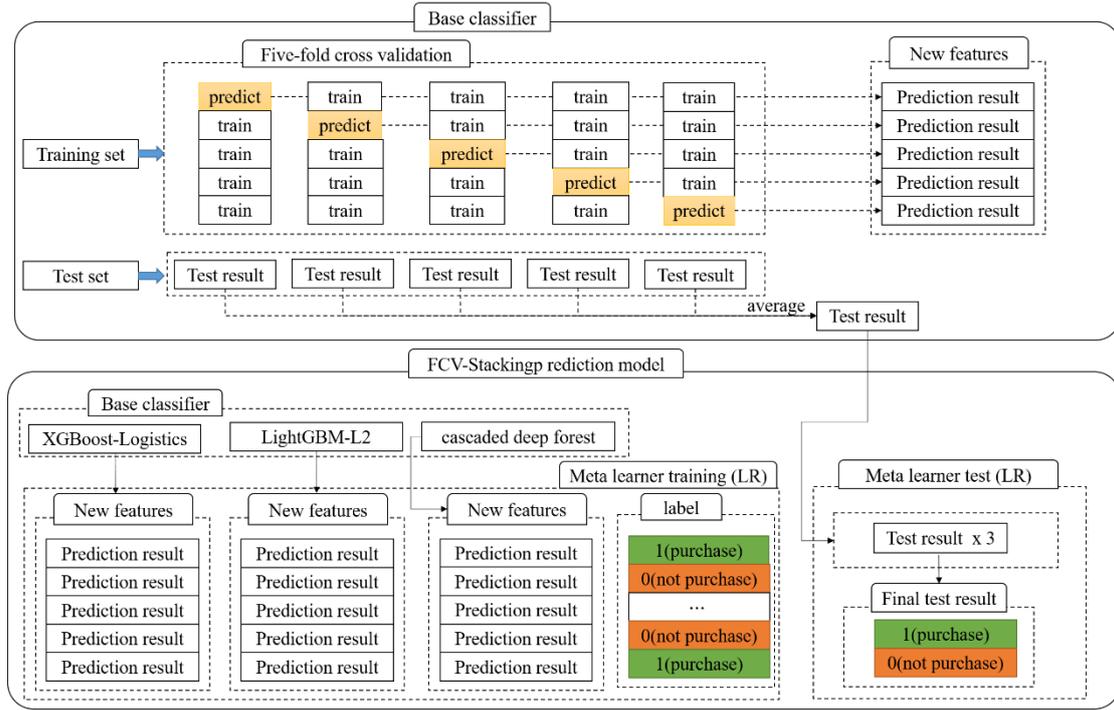


Fig. 10 FCV-Stacking ensemble learning prediction model

5.5.2 XL, LL and CDC base learners

The construction process of base learners in FCV-Stacking ensemble learning prediction model: select XL, LL and CDC base learners; divide user's historical behavioral data into training set and test set and divide the training set into five uncrossed parts: train1 to train5; select XL base learner, use train1-train4 to train the prediction model and use train5 to predict user purchasing behaviors and reserve the prediction result. The above process is repeated until train1-train5 are all predicted, and the prediction result is reserved as $B_{train}^1=(b_1, b_2, b_3, b_4, b_5)^T$; Just as XL base learner, select LL and CDC base learners to perform five-fold cross validation to obtain prediction results $B_{train}^2=(b_1, b_2, b_3, b_4, b_5)^T$ and $B_{train}^3=(b_1, b_2, b_3, b_4, b_5)^T$; During the construction process of base learners, each base learner is used to test the test set for five times, average value of the five test results is taken, and the final test results $B_{est}^1=(b_1)^T$,

$B_{est}^2=(b_2)^T$ and $B_{est}^3=(b_3)^T$ are obtained.

5.5.3 Logistic regression meta learner

As logistic regression model can mitigate Stacking overfitting risk by virtue of simple structure and strong generalization ability^[34], the meta learner of FCV-Stacking prediction model is modeled using logistic regression model as follows:

(1) According to the output $B_{train}=\{B_{train}^1, B_{train}^2, B_{train}^3\}$ of training sets of the three base learners, logistic regression equation $y=wB_{train}+b$ is constructed, where w is weight value, b is partial regression coefficient and y is a dependent variable deciding whether the user purchases.

(2) The value of dependent variable y is taken as, and the probability for user to purchase is $\rho = P(y=1 | B_{train})$, and the probability is $1 - \rho$ under $y=0$.

(3) Maximum likelihood function method is used to solve the regression coefficient in the model and evaluate the optimization model.

(4) The test value $B_{test} = \{B_{test}^1, B_{test}^2, B_{test}^3\}$ of the base learners is input to obtain and evaluate the final test values.

The detailed algorithm process is shown in Tab. 11.

Tab. 11 FCV-Stacking user purchasing behavioral prediction model

Input:	User behavior data (D) and commodity information data (P);
Output:	User-commodity pair ($U - I$), whether the commodity is purchased (1: purchased, 0: not purchased);
Process	
1:	read data = $\{D, P\}$ //Read input data
2:	// Use "sliding window" and ICIKMDs algorithm to balance sample data $date^* = balance(date)$
3	// Build user purchasing behavioral features create $V_j = \{v_1, v_2, \dots, v_{204}\}$
4:	// Use XGBSFS algorithm to select features $V_j^* = XGBSFS(V_j)$
5:	// Construct and train XL base learner $B_{train}^1 = XL \text{ Classifier}(date^*(V_j^*))$
6:	// Construct and train LL base learner $B_{train}^2 = LL \text{ Classifier}(date^*(V_j^*))$
7:	// Construct and train CDC base learner $B_{train}^3 = DC \text{ Classifier}(date^*(V_j^*))$
8:	// Construct logistic regression meta learner, and output the five-fold cross validation results of training sets of the three base learners for integrated model training $log = logistic(data(train_out(5 \times 3)))$
9:	// Input the results output by test sets of the three base learners into logistic regression meta learner for model integrated prediction $buy = log(data(test_out(1 \times 3)))$

6. Experiment and analysis

6.1 Experimental setting and evaluation indexes

Python programming language was used to implement all models, the main parameters of

which are shown in Tab. 12.

Tab. 12 Main parameter settings of the models

模型类别 Model type	主要参数 Main parameter	参数值 Parameter value	
XL	booster	gbtree	
	n_estimator	500	
	learning_rate	0.05	
	gamma	0	
	subsample	0.5	
	colsample_bytree	0.8	
	max_depth	10	
	eval_metric	logloss	
	min_child_weight	6	
LL	boosting_type	gdbt	
	learning_rate	0.1	
	num_leave	50	
	max_depth	6	
	num_leaves	64	
	min_child_samples	20	
	min_child_weight	0.002	
	reg_lambda	0.03	
CDC	type of forests	completely-random tree forest, random forest	
	multi-grained scanning	forests	2
		trees in each forest	1000
		sliding window size	$\{\lfloor 151/16 \rfloor, \lfloor 151/8 \rfloor, \lfloor 151/4 \rfloor\}$
	cascade	forests	8
		trees in each forest	500

Precision (P), recall rate (R) and F1 score were used to evaluate the performance of prediction models. Real types in the example and predicted types were combined and divided into true positive (TP), false positive (FP), true negative (TN) and false negative (FN) types, and the confusion matrix is shown in Tab. 13. The model with greater value of prediction result on principal diagonal and smaller value on counter diagonal was better. After digital processing of the confusion matrix, P, R and F1 scores can be obtained, and the calculation formulas are respectively:

$$P = \frac{TP}{TP + FP} \quad (15)$$

$$R = \frac{TP}{TP + FN} \quad (16)$$

$$F_1 = \frac{2 \times P \times R}{P + R} \quad (17)$$

Tab. 13 Confusion matrix of user purchasing behaviors

Reality	Prediction result
---------	-------------------

	Positive sample (purchase)	Negative sample (not purchase)
Positive sample (purchase)	TP (purchase predicted, actual purchase)	FN (not purchase predicted, actual purchase)
Negative sample (not purchase)	FP (purchase predicted, actual not purchase)	TN (not purchase predicted, actual not purchase)

6.2 Experimental results and analysis

6.2.1 Selection of comparison models

To evaluate the performance of FCVS ensemble learning prediction model, it was compared with comparative experimental models in Tab. 14 and ablation experimental models in Tab. 15:

1) Comparative experimental models

Tab. 14 Comparative experimental models

Model name	Model description
co-EM logical regression (co-EM-LR)	Co-EM logistic regression is a new-type logistic regression model combining semi-supervised learning and multi-perspective learning. Inheriting the solubility of logistic model, it takes full advantages of compatibility of unlabeled data and multiple views with higher prediction accuracy than traditional logistic regression model ^[36] .
Gradient boosted decision tree (GDBT)	As a powerful classification model constituted by many independent decision trees, GDBT has been extensively applied to classification tasks and competitions in various fields and achieved good effects in the prediction of user purchasing behaviors ^[37,38] .
Multi-model stacking ensemble (MMSE)	MMSE is a two-layer Multi-Model Stacking Ensemble Model proposed in Literature [21]. Base learners of four different ensemble algorithms—RF, Adaboost, GDBT and XGBoost—are trained at the first layer. At the second layer, XGBoost algorithm serves as meta learner, the prediction results of four base learners after training are fused and the final prediction results are output.
Logistic regression-XGBoost(LR-XGBoost)	LR-XGBoost combines Logistic regression and XGBoost algorithm. XGBoost is taken as feature transformation for sample prediction. New feature vectors are constructed according to the prediction results of each regression tree and then input into Logistic regression model for the final prediction ^[39] .
Recurrent neural network (RNN)	As a classification model automatically extracting features, RNN has been broadly used in sequential data classification task of various fields and achieved excellent effects in the prediction of user purchasing behaviors, too ^[40,41] .
Multilayer perceptron-long and short-time memory (MLP-LSTM) network	MLP-LSTM is an online purchasing behavior prediction model proposed in Literature [18], where MLP predicts user purchasing intention by inputting user information and

	LSTM uses click-stream data to predict the probability for users to leave the website without trading.
--	--

2) Ablation experimental models

Tab. 15 FCVS ensemble learning-based ablation experimental models (“√” means that the model includes this component)

Model name	Model component			
	Five-fold cross validation (FCV)	XL	LL	CDC
FCVS without XL	√		√	√
FCVS without LL	√	√		√
FCVS without CDC	√	√	√	
Stacking		√	√	√
FCVS	√	√	√	√

6.2.2 Experimental results of comparison models

To ensure accuracy and objectivity of experimental data, each model was separately operated for 10 times on the same training dataset and test dataset, and P, R and F1 scores were solved as the final experimental results of this experiment. Tab. 16 gives the experimental results of eleventh user purchasing behavioral prediction models.

Tab. 16 P, R and F1 scores of eleven user purchasing behavioral prediction models

Evaluation index		P	R	F1 score
Comparison model	co-EM-LR	0.0734±0.0025	0.0697±0.0026	0.0715±0.0046
	GDBT	0.0822±0.0010	0.0789±0.0035	0.0808±0.0044
	MMSE	0.1221±0.0041	0.0983±0.0064	0.1089±0.0055
	LR-XGBoost	0.1104±0.0032	0.1048±0.0024	0.1075±0.0045
	RNN	0.0874±0.0047	0.0783±0.0054	0.0826±0.0041
	MLP-LSTM	0.0852±0.0019	0.0708±0.0065	0.0773±0.0021
Ablation model	FCVS without XL	0.1704±0.0054	0.1311±0.0012	0.1482±0.0042
	FCVS without LL	0.1652±0.0036	0.1458±0.0024	0.1549±0.0012
	FCVS without CDC	0.1557±0.0054	0.1326±0.0041	0.1432±0.0028
	Stacking	0.1502±0.0029	0.1835±0.0050	0.1652±0.0036
	FCVS	0.1714±0.0013	0.1871±0.0027	0.1789±0.0024

Through the experimental results of FCVS model and comparison models as shown in Tab. 16, the following observed results can be obtained.

(1) In comparison with machine learning model co-EM-LR, neural network models based on RNN and MLP-LSTM have better effects, because RNN and MLP-LSTM neural networks have very good modeling effects for sequential data in user’s historical purchasing behavioral data, and compared with co-EM-LR machine learning model, neural network models have better feature extraction and representation abilities.

(2) F1 scores of both LR-XGBoost and MMSE ensemble learning models break through 1% in the selected comparison models, and the two models achieve better effect than neural network and co-EM-LR. This is because decision tree-based learners contained in the two ensemble learning models have considerable advantages when used to process nonlinear user’s historical purchasing behavioral data. In addition, multiple decision trees-based classifiers contained in the model can collaborate with each other because of the ensemble strategy, so as to further reduce the model classification errors.

(3) F1 score of FCVS ensemble model reaches 17.89%, which is higher than that of MMSE model with the best performance among the selected models, mainly because in comparison with MMSE model, CDC base learner in FCVS ensemble model has the level-by-level learning ability similar to neural network and integrates outstanding representation ability of neural network in tree model. Furthermore, XL and LL base learners in FCVS ensemble model have better classification performance than random forest in MMSE model and Adaboost-based classifier.

Through the experimental results of FCVS ensemble model and ablation model shown in Tab. 16, the following observed results can be obtained.

(1) By comparing the experimental results of FCVS ensemble model with other ablation models, it is found that the precision, recall rate and F1 score of all ablation models are lower than those of FCVS model. The experiment verifies that all of the three base learner prediction models proposed in this paper can contribute to performance improvement of FCVS ensemble model.

(2) The F1 score of FCVS ensemble model is increased by 8.2% in comparison with Stacking ensemble model, indicating that five-fold cross validation is very beneficial for improving the prediction result of ensemble model.

(3) By comparing three models—FCVS without LL, FCVS without XL and FCVS without CDC, it can be known that the contribution degrees of three base learners LL, XL and CDC to FCVS ensemble model are decreased progressively. The experimental results suggest that the precision of CDC model when used to predict user purchasing behaviors is higher than those of XL and CDC models, because CDC model further strengthens model representation ability through cascade structure and multi-granularity scanning strategy based on giving fully play to outstanding classification ability of decision tree-based ensemble model.

6.2.3 Validation of improvement effects of three prediction models

Fig. 11 presents the changes of F1 scores before after the improvement of three prediction models LL, XL and CDC.

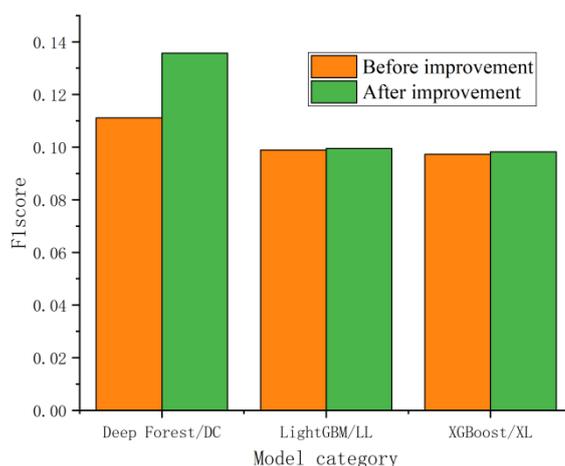


Fig. 11 Comparison of F1 scores before and after improvement of three prediction models

It can be known from the experimental results in Fig. 11 that the improved three prediction models are all superior to Deep Forest, LightGBM and XGBoost models, demonstrating the reasonability and effectiveness of the improvements of the three models, where cascaded weighted forest is improved considerably relative to the original deep forest, verifying the feasibility of

improving model performance by solving sparse connectivity problem of deep forest.

6.2.4 Influence features of purchasing behaviors

To make it convenient for analyzing influence features of user purchasing behaviors, Tab. 17 and Tab. 18 present influence features ranking the first 10 and last 10 in the XGBSFS feature selection algorithm.

Tab. 17 Statistics of influence features ranking top 10 in XGBSFS feature selection algorithm

Feature ranking	Corresponding feature
1	Purchasing behavior quantity of one user for one commodity 1 day before the prediction date
2	Quantity of one commodity added to cart by one user 1 day before the prediction date
3	Browsing behavior quantity of one user for one commodity 1 day before the prediction date
4	Purchasing behavior quantity of one user for one commodity 3 days before the prediction date
5	Ratio of quantity of one commodity purchase by one user 1 day before the prediction date to total quantity of the commodity purchased 1 day before the prediction date
6	Time span from one user purchases one commodity the first time until the last purchasing time
7	Ratio of quantity of one commodity purchased by one user 3 days before the prediction date to total quantity of the commodity purchased 3 days before the prediction date
8	Ranking of purchasing behavior quantity of one user for one commodity in the quantity of the user's all commodity purchasing behaviors
9	Quantity of one commodity added to cart by one user 3 days before the prediction date
10	Purchasing behavior quantity of one user 1 day before the prediction date

From behavior types involved in features as seen in Tab. 17, 7 among the features ranking top 10 are about purchasing behaviors, where user purchasing behavior for one commodity is involved in features (1), (4), (7) and (8), meaning that the importance of user's historical purchasing behaviors to user purchasing behavioral prediction is higher than historical browse, collect and add to cart behaviors. In the meantime, it can be known from features (2) and (9) that user's historical add to cart behavior also exerts an important effect on the prediction result. As for the time involved in the features, most of the features ranking top 10 are about user purchasing behaviors 1 day before the prediction date, e.g. features (1), (2), (3), (5) and (10). In addition, time-related features (4), (7) and (9) are also about user purchasing behaviors 3 days before the prediction date, indicating that the closer the user's historical purchasing behavior to the prediction date, the greater the influence on prediction of user purchasing behavior. The proportional features (5) and (7) of quantity of one commodity purchased by one user to total quantity of commodity purchased, and historical purchase life cycle (6) of one user for one commodity also have a certain effect on user purchasing behavioral prediction.

Tab. 18 Statistics of influence features ranking the last 10 in XGBSFS feature selection algorithm

Feature ranking	Corresponding feature
1	Ratio of number of days with behavior records of one user to total number of days
2	Ratio of purchasing behavior quantity of one user 6 days before the prediction date to quantity of collect behaviors
3	Ratio of purchasing quantity of one commodity classification 6 days before the prediction date to quantity of the commodity classification collected
4	Quantity of collect behaviors of one user 6 days before the prediction date
5	User quantity of one commodity classification 6 days before the prediction date
6	Quantity of one commodity classification collected by one user 6 days before the prediction date
7	Behavior quantity of one user 6 days before the prediction date
8	User behavior quantity for one commodity classification 6 days before the prediction date
9	Time span from one user collects one commodity classification until the last time
10	Browsing behavior quantity of one commodity classification 6 days before the prediction date

It can be known from rankings in Tab. 18 that in terms of behavior types involved in the features, 4 among the features ranking the last 10 are about collect behaviors, respectively being features (2), (3), (4) and (6). This indicates that the influence of user historical collect behavior on user purchasing behaviors is weaker than those of historical browsing, purchasing and add to cart behaviors. From time involved in the features, 8 among the features ranking the last 10 are about user purchasing behaviors 6 days before the prediction date, accounting for 4/5 of total behavior quantity, suggesting that user historical purchasing behavior distant from the prediction date has a minor effect on prediction of purchasing behaviors, and this further verifies the feasibility and scientificity of the 7-day “time window” set in this paper. It is noteworthy that the core index for M-Commerce platforms, namely purchasing conversion ratio, appears in the influence features ranking the last 10, ranking the last two or three places. Therefore, purchasing conversion ratio feature describes the probability for one user to purchase one commodity, but it has less effect on prediction of user purchasing behaviors.

6.2.5 Validation of sample balance method

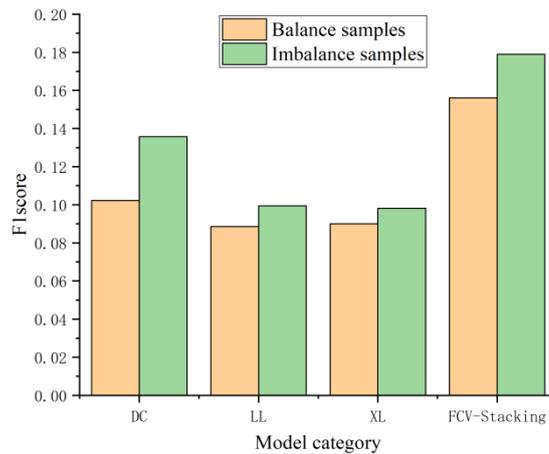


Fig. 12 Comparison of F1 scores of the four models before and after sample balance

As shown in Fig. 15, processed through “sliding window” and ICIKMDS sample balance algorithm, F1 scores of LL, XL, CDC and FCVS are elevated to different degrees, where those of CDC and FCVS models are evidently increased. The experimental results show that the precision of the prediction model can be improved by under-sampling of negative samples within “sliding window” using ICIKMDS algorithm while increasing the quantity of positive samples using “sliding window”. This is because when the quantity of positive samples is increased by “sliding window” method, the used ICIKMDS algorithm can perform under-sampling of negative samples in each “time window” according to sequential data of user historical behaviors in each segment of “time window”. Moreover, it largely keeps the original characteristics of data samples and relieve the “data skew” problem of traditional “under-sampling” method due to data loss, thus reducing the probability of overall “skew” of the model.

7. Discussion

The user purchasing behavior prediction problems in M-Commerce platforms under wireless communication network environment were mainly investigated in this paper. Sample balance, feature engineering and ensemble learning were introduced for modeling of user historical behavioral data to solve the prediction problem of user future purchasing behaviors. To be specific, user historical behavior data of sample balance were acquired by combining “sliding window”-centroid under-sampling with sample balancing method. After then, influence features of user purchasing behaviors from three perspectives—user, commodity and interaction—were established through data analysis, and meanwhile, feature selection was operated using XGBSFS algorithm. In the end, ensemble learning-based prediction model FCVS was proposed, which effectively fused three prediction models XL, LL and CDC to realize the prediction of user purchasing behaviors. The proposed model was validated using real user historical behavioral datasets on Alibaba M-Commerce platform, and the experimental results verified the effectiveness and superiority of the proposed method in the aspect of user purchasing behavioral prediction.

The emphasis will be laid on the effects of user historical ratings and sentiment polarity of comments on user purchasing behaviors in the future research work. The two factors will be included into the feature engineering to construct a prediction model and thus further improve the accuracy of user purchasing behavioral prediction.

List of abbreviations

(CDC)

Control Data Corp

Declarations

Availability of data and material

The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Competing interests

declares that he has no conflict of interest.

* Research involving human participants and/or animals

Ethical approval: This article does not contain any studies with human participants or animals performed by any of the authors.

* Informed consent

All authors agree to submit this version and claim that no part of this manuscript has been published or submitted elsewhere.

Funding

None

Authors' contributions

Huibing Zhang is responsible for collecting the receipt of the experiment, Junchao Dong is responsible for the simulation and simulation of the experiment

Acknowledgements

None

References:

- [1] Da'u A, Salim N. Recommendation system based on deep learning methods: a systematic review and new directions[J]. *Artificial Intelligence Review*, 2019: 1-40.
- [2] Bi J W, Liu Y, Fan Z P. A deep neural networks based recommendation algorithm using user and item basic data[J]. *International Journal of Machine Learning and Cybernetics*, 2020, 11(4): 763-777.
- [3] Zhao Y, Yao L, Zhang Y. Purchase prediction using Tmall-specific features[J]. *Concurrency and computation: practice and experience*, 2016, 28(14): 3879-3894.
- [4] Li X, Qian S, Peng F, et al. Deep convolutional neural network and multi-view stacking ensemble in ali mobile recommendation algorithm competition: The solution to the winning of ali mobile recommendation algorithm[C]//2015 IEEE International Conference on Data Mining

- Workshop (ICDMW). IEEE, 2015: 1055-1062.
- [5] Li Q, Gu M, Zhou K, et al. Multi-Classes Feature Engineering with Sliding Window for Purchase Prediction in Mobile Commerce[C]//2015 IEEE International Conference on Data Mining Workshop (ICDMW). IEEE, 2015: 1048-1054.
- [6] Li D, Zhao G, Wang Z, et al. A method of purchase prediction based on user behavior log[C]//2015 IEEE International Conference on Data Mining Workshop (ICDMW). IEEE, 2015: 1031-1039.
- [7] Ke G, Meng Q, Finley T, et al. Lightgbm: A highly efficient gradient boosting decision tree[C]//Advances in neural information processing systems. 2017: 3146-3154.
- [8] Lei C, Dai H, Yu Z, et al. A service recommendation algorithm with the transfer learning based matrix factorization to improve cloud security[J]. Information Sciences, 2020, 513: 98-111.
- [9] Bobadilla J, Alonso S, Hernando A. Deep Learning Architecture for Collaborative Filtering Recommender Systems[J]. Applied Sciences, 2020, 10(7): 2441.
- [10] Mongia A, Jhamb N, Chouzenoux E, et al. Deep latent factor model for collaborative filtering[J]. Signal Processing, 2020, 169: 107366.
- [11] Nassar N, Jafar A, Rahhal Y. A novel deep multi-criteria collaborative filtering model for recommendation system[J]. Knowledge-Based Systems, 2020, 187: 104811.
- [12] Pajuelo-Holguera F, Gómez-Pulido J A, Ortega F, et al. Recommender system implementations for embedded collaborative filtering applications[J]. Microprocessors and Microsystems, 2020: 102997.
- [13] Wu Y, Ngai E W T, Wu P, et al. Fake online reviews: Literature review, synthesis, and directions for future research[J]. Decision Support Systems, 2020: 113280.
- [14] Dong Y, Jiang W. Brand purchase prediction based on time-evolving user behaviors in e-commerce[J]. Concurrency and Computation: Practice and Experience, 2019, 31(1): e4882.
- [15] Tang L, Wang A, Xu Z, et al. Online-purchasing behavior forecasting with a firefly algorithm-based SVM model considering shopping cart use[J]. Eurasia Journal of Mathematics, Science and Technology Education, 2017, 13(12): 7967-7983.
- [16] Sakar C O, Polat S O, Katircioglu M, et al. Real-time prediction of online shoppers' purchasing intention using multilayer perceptron and LSTM recurrent neural networks[J]. Neural Computing and Applications, 2019, 31(10): 6893-6908.
- [17] Guo L, Hua L, Jia R, et al. Buying or Browsing?: Predicting Real-time Purchasing Intent using Attention-based Deep Network with Multiple Behavior[C]//Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2019: 1984-1992.
- [18] Song H S. Comparison of Performance between MLP and RNN Model to Predict Purchase Timing for Repurchase Product[J]. Journal of Information Technology Applications and Management, 2017, 24(1): 111-128.
- [19] Ling C, Zhang T, Chen Y. Customer Purchase Intent Prediction Under Online Multi-Channel Promotion: A Feature-Combined Deep Learning Framework [J]. IEEE Access, 2019, 7: 112963-112976.
- [20] Chen W, Li Z, Zhang M. Linear and non-linear models for purchase prediction[M]//Proceedings of the 2015 International ACM Recommender Systems Challenge. 2015: 1-4.
- [21] Zhou A, Ren K, Li X, et al. MMSE: A Multi-Model Stacking Ensemble Learning Algorithm for Purchase Prediction[C]//2019 IEEE 8th Joint International Information Technology and

Artificial Intelligence Conference (ITAIC). IEEE, 2019: 96-102.

[22] Hou C, Chen C, Wang J. Tree-Based Feature Transformation for Purchase Behavior Prediction[J]. IEICE TRANSACTIONS on Information and Systems, 2018, 101(5): 1441-1444.

[23] <https://tianchi.aliyun.com/competition/entrance/231522/introduction>

[24] Faris H, Heidari A A, Ala'M A Z, et al. Time-varying hierarchical chains of salps with random weight networks for feature selection[J]. Expert Systems with Applications, 2020, 140: 112898.

[25] Chen T, Guestrin C. Xgboost: A scalable tree boosting system[C]//Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. 2016: 785-794.

[26] Ke G, Meng Q, Finley T, et al. Lightgbm: A highly efficient gradient boosting decision tree[C]//Advances in neural information processing systems. 2017: 3146-3154.

[27] Artacho F J A, Campoy R, Elser V. An enhanced formulation for solving graph coloring problems with the Douglas–Rachford algorithm[J]. Journal of Global Optimization, 2020: 1-21.

[28] Zhou J, Zhao X, Zhang X, et al. Task Allocation for Multi-Agent Systems Based on Distributed Many-Objective Evolutionary Algorithm and Greedy Algorithm[J]. IEEE Access, 2020, 8: 19306-19318.

[29] Zhou Z H, Feng J. Deep forest[J]. arXiv preprint arXiv:1702.08835, 2017.

[30] Liu H, Duan Z, Chen C. A hybrid multi-resolution multi-objective ensemble model and its application for forecasting of daily PM_{2.5} concentrations[J]. Information Sciences, 2020, 516: 266-292.

[31] Jiang M, Liu J, Zhang L, et al. An improved Stacking framework for stock index prediction by leveraging tree-based ensemble models and deep learning algorithms[J]. Physica A: Statistical Mechanics and its Applications, 2020, 541: 122272.

[32] Li Y, Yang Z, Chen X, et al. A stacking model using URL and HTML features for phishing webpage detection[J]. Future Generation Computer Systems, 2019, 94: 27-39.

[33] Wang Y, Wang D, Geng N, et al. Stacking-based ensemble learning of decision trees for interpretable prostate cancer detection[J]. Applied Soft Computing, 2019, 77: 188-204.

[34] Zhu G, Wu Z, Wang Y, et al. Online purchase decisions for tourism M-Commerce [J]. Electronic Commerce Research and Applications, 2019, 38: 100887.

[35] Chen C, Hou C, Xiao J, et al. Enhancing purchase behavior prediction with temporally popular items[J]. IEICE TRANSACTIONS on Information and Systems, 2017, 100(9): 2237-2240.

[36] Martínez A, Schmuck C, Pereverzyev Jr S, et al. A machine learning framework for customer purchase prediction in the non-contractual setting[J]. European Journal of Operational Research, 2020, 281(3): 588-596.

[37] XingFen W, Xiangbin Y, Yangchun M. Research on user consumption behavior prediction based on improved XGBoost algorithm[C]//2018 IEEE International Conference on Big Data (Big Data). IEEE, 2018: 4169-4175.

[38] Chu Y, Yang H K, Peng W C. Predicting Online User Purchase Behavior Based on Browsing History[C]//2019 IEEE 35th International Conference on Data Engineering Workshops (ICDEW). IEEE, 2019: 185-192.

[39] Sheil H, Rana O, Reilly R. Predicting purchasing intent: Automatic feature learning using recurrent neural networks[J]. arXiv preprint arXiv:1807.08207, 2018.

Figure 1:

Title: Commodity purchase conditions every day within one month

Legend: Figure 1 show Commodity purchase conditions every day within one month

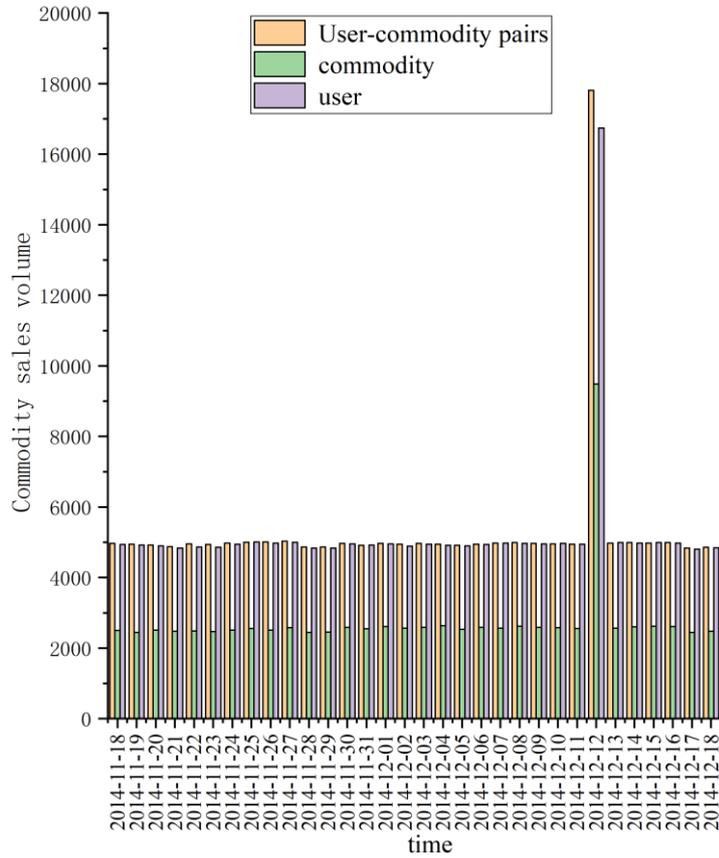


Figure 2:

Title: Time difference-dependent quantity changes of three behaviors and purchasing behavior

Legend: Figure 2 show Time difference-dependent quantity changes of three behaviors and purchasing behavior

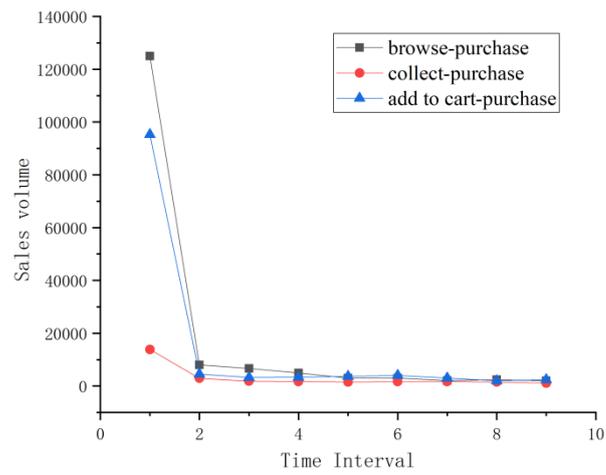
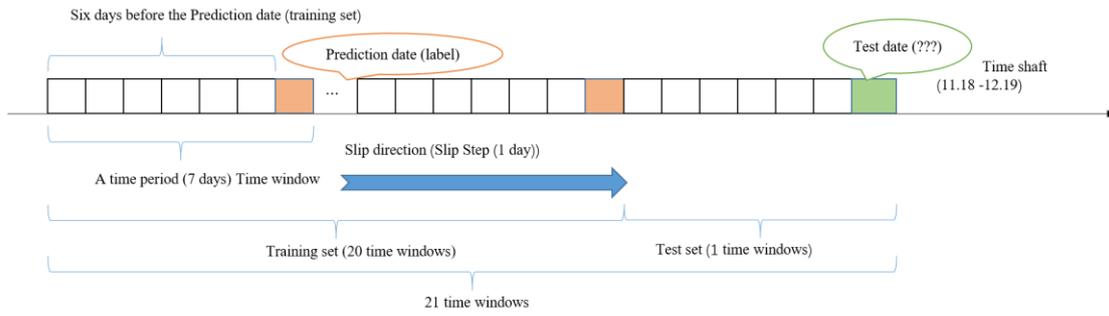


Figure 3:

Title: Construction process of training set and test set through “sliding window” method

Legend: Figure 3 show Construction process of training set and test set through “sliding window” method



Slide number	Training set						label
1	11.18	11.19	11.20	11.21	11.22	11.23	11.24
2	11.19	11.20	11.21	11.22	11.23	11.24	11.25
3	11.20	11.21	11.22	11.23	11.24	11.25	11.26
4	11.21	11.22	11.23	11.24	11.25	11.26	11.27
5	11.22	11.23	11.24	11.25	11.26	11.27	11.28
6	11.23	11.24	11.25	11.26	11.27	11.28	11.29
7	11.24	11.25	11.26	11.27	11.28	11.29	11.30
8	11.25	11.26	11.27	11.28	11.29	11.30	11.31
9	11.26	11.27	11.28	11.29	11.30	11.31	12.1
10	11.27	11.28	11.29	11.30	11.31	12.1	12.2
11	11.28	11.29	11.30	11.31	12.1	12.2	12.3
12	11.29	11.30	11.31	12.1	12.2	12.3	12.4
13	11.30	11.31	12.1	12.2	12.3	12.4	12.5
14	11.31	12.1	12.2	12.3	12.4	12.5	12.6
15	12.1	12.2	12.3	12.4	12.5	12.6	12.7
16	12.2	12.3	12.4	12.5	12.6	12.7	12.8
17	12.3	12.4	12.5	12.6	12.7	12.8	12.9
18	12.4	12.5	12.6	12.7	12.8	12.9	12.10
19	12.5	12.6	12.7	12.8	12.9	12.10	12.11
20	12.6	12.7	12.8	12.9	12.10	12.11	12.12
Test set							result
12.13	12.14	12.15	12.16	12.17	12.18	12.19	

Figure 4:

Title: Data format of each row in the new data sheet

Legend: Figure 4 show Data format of each row in the new data sheet

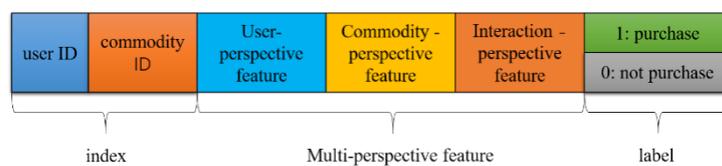


Figure 5:

Title: Framework of user purchasing behavioral prediction model

Legend: Figure 5 show Framework of user purchasing behavioral prediction model

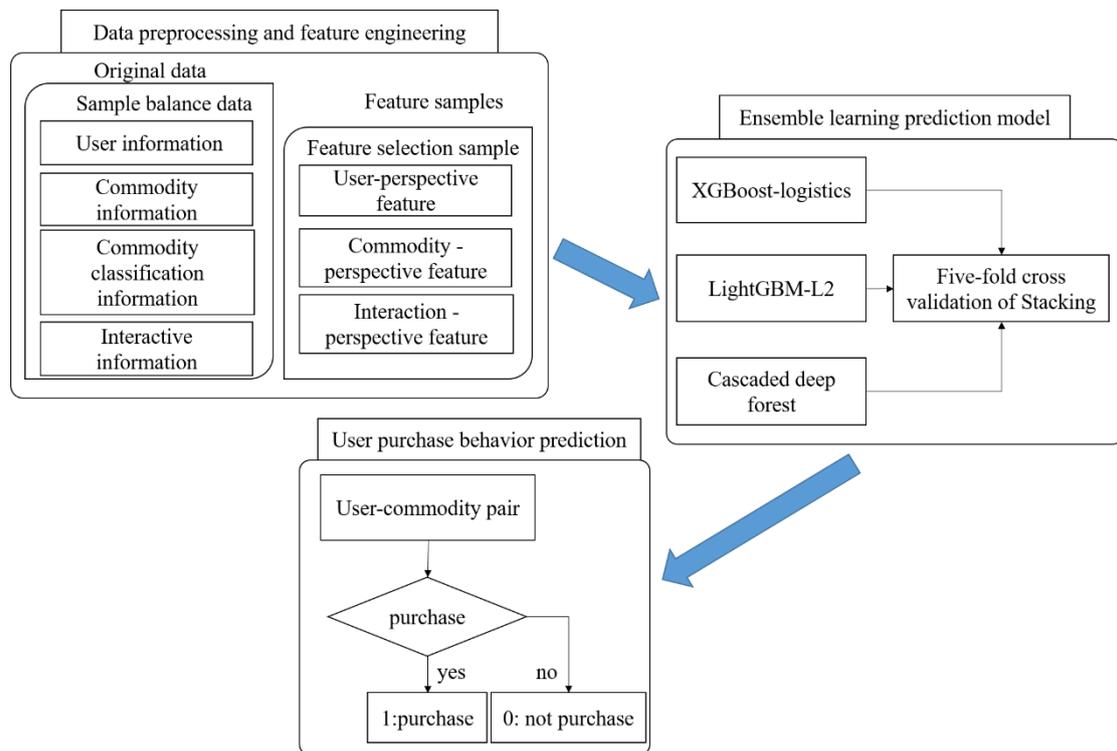


Figure 6:

Title: Structural illustration of cascaded deep forest

Legend: Figure 6 show Structural illustration of cascaded deep forest

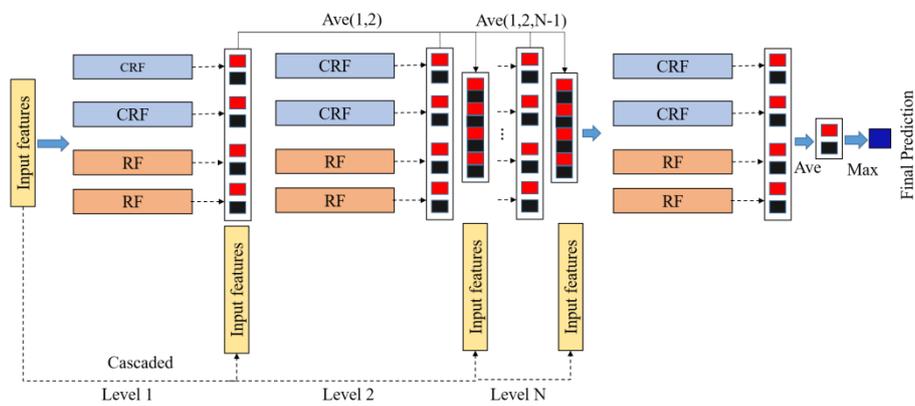


Figure 7:

Title: Class vector generation

Legend: Figure 7 show Class vector generation

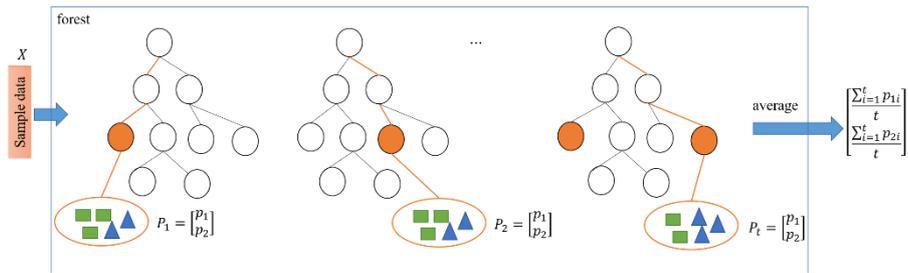


Figure 8:

Title: Scanning strategy of one “sliding window”

Legend: Figure 8 show Scanning strategy of one “sliding window”

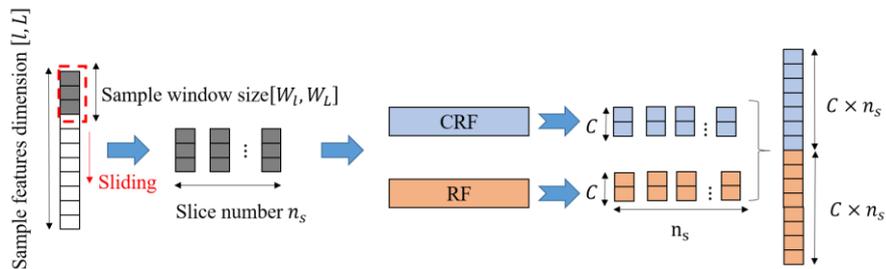


Figure 9:

Title: Cascaded deep forest processing of multi-granularity scanning

Legend: Figure 9 show Cascaded deep forest processing of multi-granularity scanning

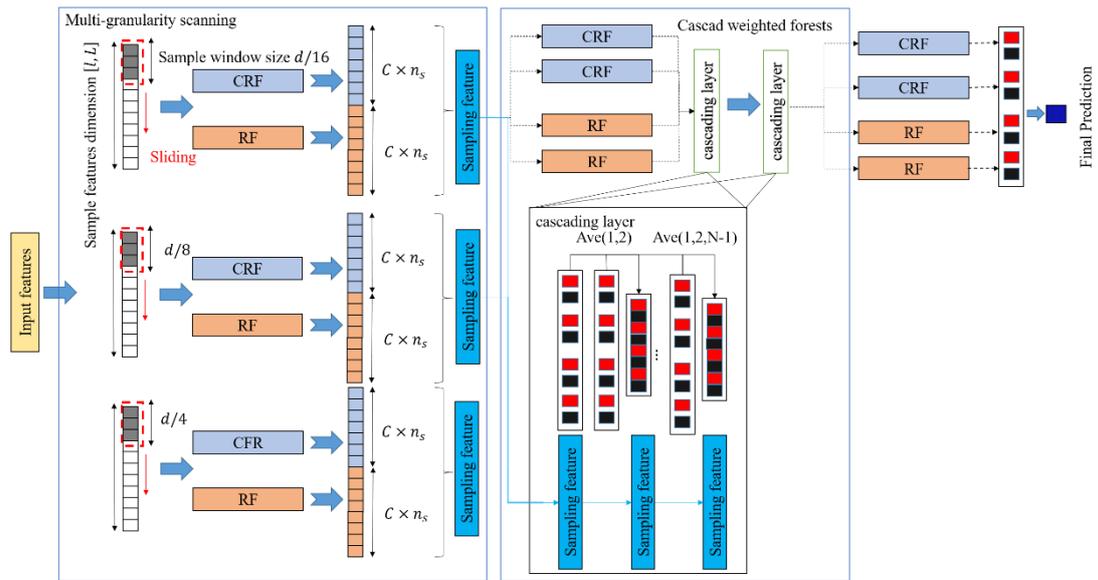


Figure 10:

Title: FCV-Stacking ensemble learning prediction model

Legend: Figure 10 show FCV-Stacking ensemble learning prediction model

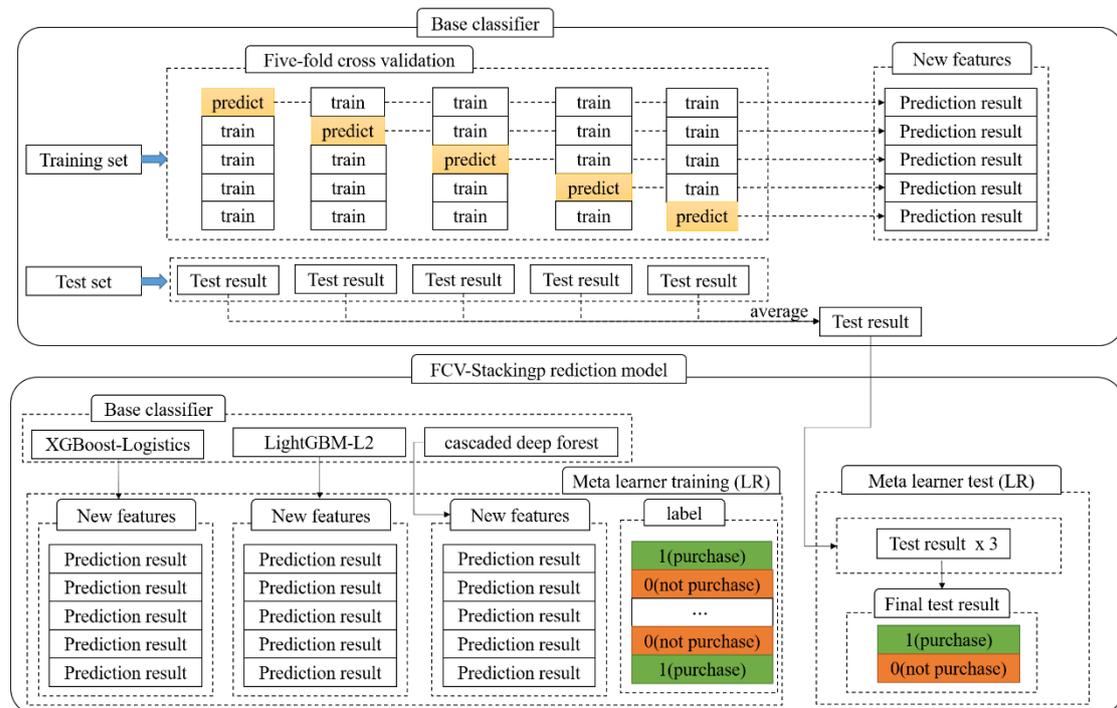


Figure 11:

Title: Comparison of F1 scores before and after improvement of three prediction models

Legend: Figure 11 show Comparison of F1 scores before and after improvement of three prediction models

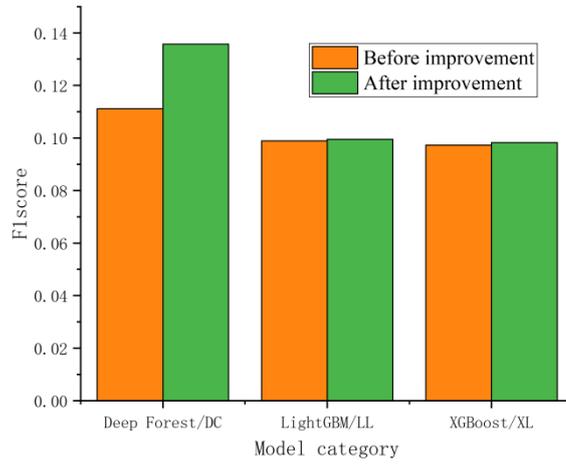


Figure 12:

Title: Comparison of F1 scores of the four models before and after sample balance

Legend: Figure 12 show Comparison of F1 scores of the four models before and after sample balance

