Detection of Merchant Category Codes Application Based on Root-seeking Fast Hierarchical Clustering Algorithm

Long Wang

Yiwu Natural Resources and Planning Bureau, Yiwu, China

356479632@qq.com

Keywords: data mining; hierarchical clustering; merchant category code application; industry model

Abstract: The research on the application detection algorithm of merchant category code is carried out with data mining technology. Based on the data mining technology, the basic innovation work is carried out, and a new clustering algorithm – “fast hierarchical clustering algorithm based on root-seeking” (RS algorithm) is proposed. A comparative experiment is performed for the algorithm. Experimental results show that RS algorithm is superior to other classical algorithms in most data sets. Then a complete set of merchant category code application detection model is proposed. Five classification algorithms are selected and tested on four data sets. The experimental results show that the proposed merchant category code application detection model is feasible, and its accuracy rate and recall rate on three data sets are above 80%. To sum up, theoretical innovation is carried out with data mining technology, and a “merchant category code application detection model” is proposed based on the application scenario of data mining in the financial field. The effectiveness of the proposed algorithm and model is verified by a lot of experiments.

1. Introduction

With the continuous development of e-commerce, people no longer carry large amounts of cash when they go out and pay more on credit card, which drives more and more merchants to prepare Point Of Sale (POS) machines. POS machines (point-of-sale intelligence management system) are mainly provided by China Unionpay Center. Merchants can simply use the POS machine to read the information of bank card, the cardholder input the corresponding password to get real-time transfer payment and print detailed notes. POS machine provides merchants with fast and convenient operation. It realizes the integration of credit card, debit card and other bank cards, without the tedious steps of manual input of card number information, which not only ensures the accuracy but also improves the efficiency. However, to regulate the use of POS machines, the rates charged by the charging agencies in using POS machines in different industries are also different.

The third-party payment institution shall prepare a category code - the merchant category code (MCC) for the merchant according to the main business income of the industry operated by the merchant. That is, the number of the 8th to 11th digit on the receipt printed by POS when payment is made with the card, the service charge proportion of each merchant to pay for the card is determined according to this category code. POS machines are generally handled by third-party agencies. Therefore, the popularity of POS machines has led to the increase of third-party agencies, which has led to the increasingly fierce competition among third-party agencies. To attract more merchants to apply for POS machines, third-party institutions may falsify merchant categories, modify merchant names and other information, or forge merchant online materials to reduce the service charge, so as to achieve the purpose of illegal application of low zero deduction rate.

For example, if you buy a movie ticket at a movie theater by card, the merchant will have to pay a service charge of 1.25% of the total amount. However, if the merchant prepares a POS machine of a household appliance store, the merchant only needs to pay a service charge of 0.38% or less of the total amount, which is a large amount of income for a merchant with a large transaction volume. This kind of phenomenon is “application merchant category code”. According to relevant reports,

DOI: 10.25236/icscbd.2018.012

the relevant departments of China UnionPay found that in the first half of 2014, China had confirmed more than 180,000 illegal “application merchant category code” merchants, accounting for 40% of the total number of illegal merchants. Among the 40% of merchants, 80 % of them are illegal merchants that use low-fee merchant category codes.

Merchant category code is set up by the receiving organization for the contracted merchants, which is used to indicate the trading environment of Unionpay card, the main business scope and industry affiliation of the merchant. It is the main basis for judging the settlement fee standard of domestic multi-bank transaction merchants. It is also one of the important basic data for the industry analysis and report of Unionpay card transaction and the risk management and control of Unionpay card business. Merchant category code consists of four digit code. When signing a contract with a merchant, the receiving institution shall investigate and understand the actual main business, business scope and business status of the merchant, and ensure that the merchant has good business reputation and stable financial status. The bank card receipt management regulation stipulates that the merchant number (merchant code format) is 15. It consists of the organization code (3 digits) + area code (4 digits) + merchant type (4 digits) + merchant order number (4 digits). The merchant category code set by the receiving organization for the specially engaged merchant must be consistent with the main business of the merchant. Among them, the main business of merchant refers to the business type that contributes the most to the daily operating income of merchant.

There are various hazards in applying merchant category codes. For customers, different MCCs (merchant category codes) have different service charges, which also determine the income of different card issuing banks, thus affecting whether there are credits in the customer credit card. Many specific ranges of MCC codes, such as public interest MCC codes, do not accumulate credits. This is a loss for the customer. In addition, if the reimbursement voucher is needed, the customer clearly consumes in the catering industry, but the voucher shows that the consumption is in the public welfare. Of course, there will be great inconvenience in reimbursement. What’s more, if the card issuer finds abnormal consumption in the detection of credit card fraud, it may reduce the credit card limit or even stop using the credit card. In addition, the POS machines used in violation are likely to be installed with software to steal the information of consumers’ bank cards.

To sum up, to make the merchant category code better play the above role, whether from the perspective of market supervision or from the perspective of consumers, it is necessary to maintain the good order of the acquiring market and provide the basis for the electronic payment to provide people with faster, better and more secure services.

2. Literature review

Since the development of the banking industry, the banking system has been continuously mature, accompanied by a number of lawbreakers have also produced a large number of frauds. Relative to the domestic market, the foreign banking system has started very early, so it will be more perfect and more mature. The research on the recognition of fraud in foreign countries is relatively early and profound, and the research in China has gradually matured in recent years. Next, the research on fraud detection of credit cards will be introduced one by one.

In the study of credit card fraud, the neural network algorithm has been widely used because of its high recognition rate, good robustness and strong implementation. In 1994, Sushmito Ghosh used the radial basis function (RBF) neural network to detect credit card fraud and put it into use at bank of Mellon. Aleskerov and Freisleben proposed a data mining system (CARDWATCH) that USES historical transaction data to build a neural network model to detect fraud. Alex proposed a neural network model based on merchant credit and ROC analysis [1]. Carneiro proposed a credit card fraud detection system recognized by a cascade neural network. That is, the Gating Network (GNS) was used to aggregate the confidence values of three parallel artificial neural network classifiers, and the weight of the gating network was optimized by the Imperialist Competition Algorithm (ICA). Experiments showed that this algorithm can obtain very high recognition rate and reliability [2]. Dal used an evolutionary approach from adaptively constructing neural network models to identify fraud [3]. Sam and Karl combined Bayesian and neural network to build a detection model. They found that simple Bayesian networks spent less time and were more accurate in building the classifier, but it didn't work well with the new data. So they added a neural network algorithm, and the effect was improved. Patil's Bayesian network approach combined rule-based filtering, evidence theory (Dempster–Shafertheory), and introduced the concept of suspect score [4]. Bahnsen used Bayesian algorithm to build an experimental system of bank anti-fraud model, so as to identify whether credit card users have committed fraud [5].

The transaction record of credit card also has the time characteristic, its record has the corresponding correlation. Therefore, the association rule can be used to find out the rule of customer transaction and then combine with the decision tree for anomaly detection. Kokkinak used the decision tree to analyze the differences between normal and abnormal transactions, and then used the differences to identify fraud. Adewumi introduced new machine learning theories of Cost Sensitivity into decision trees. Moreover, he found that the traditional performance of this decision tree algorithm, such as accuracy rate and recall rate, was higher than the existing algorithm, and the cost sensitivity of the new definition of credit card fraud was also very good [6].

Van proposed a multi-kernel Support Vector Machine (SVM) algorithm and introduced user configuration information instead of simple transaction information [7]; Lafond Lapalme used Hidden Markov Models (HMM) to detect fraud [8]; Chattopadhyay used this unsupervised peer group analysis method to monitor abnormal transactions [9]. Siless used new ideas of fuzzy logic to detect fraud. Firstly, the initial credit of each transaction was calculated through the first-order Sugeno fuzzy model. If the transaction was found to be suspicious, its backside credit used previously suspected scores and was calculated by applying Fuzzy-Bayesian Inferencing [10]. Brause and his peers also build fraud detection systems by combining neural networks with association analysis. Neamatollahi introduced a new algorithm based on root-seeking fast hierarchical clustering algorithm in the study of merchant class application detection algorithm and conducted the classification experiment on the whole merchant category code, achieving good results [11].

3. Fast hierarchical clustering algorithm based on root-seeking

The core of bottom-up hierarchical clustering algorithm is how to merge subclusters. This is the most critical step of the whole algorithm, because once the subclusters are merged together, the next algorithm will be based on the newborn clusters. Completed mergers can't be undone and samples between clusters can't be exchanged. The quality of the clustering results may be reduced if the proper combination is not carried out at a certain step of the clustering process. Therefore, the merging of clusters in traditional hierarchical clustering algorithm is usually based on a lot of calculation, which makes the implementation efficiency of hierarchical clustering algorithm low. To improve the efficiency of hierarchical clustering, a lot of improved methods are proposed. A new algorithm based on root-seeking fast hierarchical clustering (RS algorithm) is proposed. The RS algorithm uses greedy strategy to search for the nearest neighbors in an iterative manner to find the core points (called root nodes) located in the data-intensive region. At the same time, the traversed points are connected to establish a subtree to complete the clustering.

3.1 Algorithm thought

The fast hierarchical clustering algorithm based on root-seeking is based on the following assumptions: a data set can be divided into many clusters, and there is one core point in each cluster, which can represent other points of the cluster in which it resides. This core point is not necessarily the centroid of the cluster, it exists in areas where data is densely distributed. The "data-intensive area" here refers to the surrounding neighborhood relative to this area, and the distance between the data points in this area is the closest. The core point of cluster is called "root node". The root node can be found in data-intensive area by following steps: step 1: randomly select a node labeled C in the entire data set and construct a new linked list labeled L. Take node C as the head node of linked list L; step 2: traverse the neighbors of node C, find the node closest to node C and mark it as P. Take node P as the father node of node C; step 3: if node P is already in the linked list L then node

C at this time is taken as the root node and is denoted as R. Otherwise, mark node P as C and return to step 2.

By optimizing the stability and transitivity of root-seeking process, the root-seeking algorithm has a stable tree structure, and the clustering effect has been improved obviously. According to the above optimization in one layer clustering, the definition of root node is modified as formula (1). Among them, S_0 is the random starting point; R represents the root node searched through S_0; N(*) is the nearest neighbor search function (According to different data types, different nearest neighbor search functions need to be selected. The execution efficiency of nearest neighbor search functions will determine the execution efficiency of the whole clustering process).

$$R = \frac{\overbrace{N(\ldots N(N(S_0)))}^{k} + \overbrace{N(\ldots N(N(S_0)))}^{k-1}}{2} \tag{1}$$

After improving the stability error and transitivity error, there are many changes in RS algorithm. For example, when creating a root node, it needs to calculate the centroid; the height of the tree needs to be checked during the process of building the tree, and extra processing should be performed on the part beyond the height of the tree. The complete RS algorithm is described as follows:

Step 1: put all data nodes in the data set into the candidate set;

Step 2: if the candidate set is not empty, randomly select a node to mark as C in the candidate set, and build a new linked list to mark as L, and take node C as the head node of linked list L. Otherwise skip to step 7;

Step 3: traverse the neighborhood of node C and find the node closest to node C and marked as P. Take node P as the father node of node C;

Step 4: if the node P is already in the linked list L, establish a tree T and create the root node R of the new tree T, and the root node R is defined as the centroid of the node C and the node P at this time. Take node C as the son of R, add list L to tree T as a branch, and then return to step 2. Otherwise, continue to the next step;

Step 5: if node P is the node on a tree T, add linked list L as a branch to tree T. According to the depth limit parameter h of the tree, the nodes exceeding the depth of the tree T are sequentially deleted from the head node of the linked list L and the deleted nodes are taken as the new root node, and then return to the step (2). Otherwise, continue to the next step;

Step 6: if the length of the linked list L exceeds the depth limit parameter h of the tree at this time, the head node of the linked list L is removed from L and the node is regarded as a new root node, the node P is marked as C and then returned to step 3;

Step 7: if the number of root nodes at this time is greater than 1, put all the root nodes together as a new candidate set, and then return to step 2. Otherwise, the algorithm ends.

3.2 Comparative experiments

The comparison experiment is conducted with 14 open UCI data sets that can be downloaded via the Weka Note website. These 14 UCI datasets have a wide range of domain and a large number of data characteristics. They are all data sets with continuous numerical data and category labels. Because the data sets used for the experiments are all labeled, this algorithm is evaluated with the Fowlkesand Mallows (FM) indicator and the average accuracy (AA). According to the real label and prediction data relationship table shown in Table 1, the FM indicator can be defined as formula (1); average accuracy (AA) is defined as the average of positive accuracy (PA) and negative accuracy (NA), as shown in formula (2), (3) and (4).

Table 1 The relationship between actual tags and predictive data

		Actual tags	
		Tags: True	Tags: False
Predictive data	Prediction: True	True Positive (TP)	False Positive (FP)
	Prediction: False	False Negative (FN)	True Negative (TN)

$$FM = \sqrt{\frac{TP}{TP+FP} \times \frac{TP}{TP+FN}} \tag{2}$$

$$PA = \frac{TP}{TP+FN} \tag{3}$$

$$NA = \frac{TN}{FP+TN} \tag{4}$$

$$AA = \frac{PA+NA}{2} \tag{5}$$

In the experiment, the minimum Euclidean distance is selected as the nearest neighbor algorithm N(x), as defined by equation (6):

$$N(x) = \min\{EucDis\tan c(x, x_i) | x \neq x_i\} \tag{6}$$

The two classical clustering algorithms of K-means and the unweighted pair-group method with arithmetic means (UPGMA) in hierarchical clustering are used as the comparison algorithm in the comparison experiment.

3.3 Result and discussion

From the experimental results, the clustering effect of RS algorithm is obviously better than UPGMA algorithm and K-means algorithm. The k-means algorithm is very unstable, and the clustering results obtained through K-means have large fluctuations. On the contrary, UPGMA is a very stable clustering algorithm, and the clustering results are the same every time. RS algorithm is an improved hierarchical clustering algorithm which is similar to UPGMA algorithm and has good stability. In the experiment, the standard deviation of RS algorithm is far less than that of K-means algorithm. The standard deviation of the average accuracy of the RS algorithm is 0.77, and the standard deviation of the Fowlkesand Mallows indicator is 1.53.

RS algorithm and UPGMA algorithm are both hierarchical clustering algorithms, and their clustering results have similar tree structure. The tree structure obtained by RS algorithm and UPGMA algorithm cluster is analyzed. Figure 1 shows a set of random two-dimensional data. Data points 13, 18, 19 and 20 are far away from other data points, and they are marked with shadows.

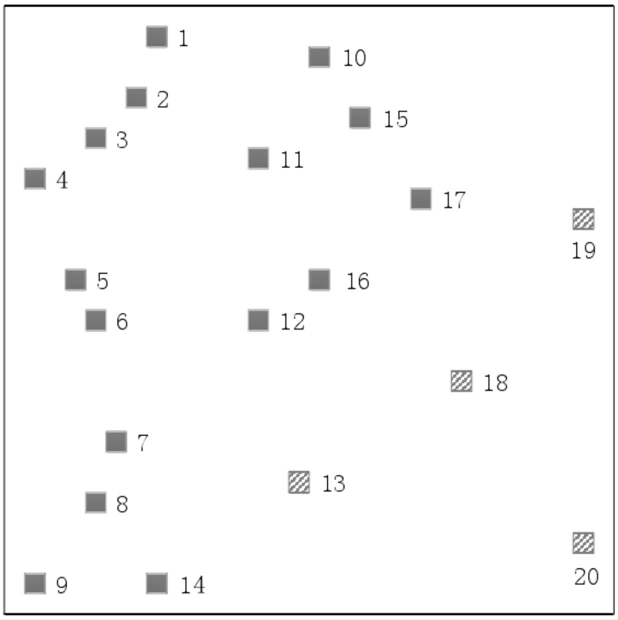

Figure. 1 Clustering comparison data legend of RS algorithm and UPGMA algorithm

The UPGMA algorithm and RS algorithm are used to cluster the data in figure 1. The tree structure obtained by UPGMA algorithm clustering is shown in figure 2. All data points are organized in the tree as leaf nodes. It can be concluded that the four isolated points far from other data points are all added to the tree in the last steps of the algorithm. In fact, such clustering results are seriously disturbed by these four isolated points. If the number of clusters is set to be any value between 2 and 5, then these isolated points will be separately divided into a cluster and become a noise cluster.

Figure 3 shows the tree structure obtained by RS algorithm clustering. Unlike the UPGMA algorithm, the RS algorithm first organizes all data nodes into appropriate subtree structures, and then further organizes the root nodes of these subtrees into the entire tree structure. In this way, the four isolated points are divided into the nearest subtree, thus avoiding the interference of noise points.

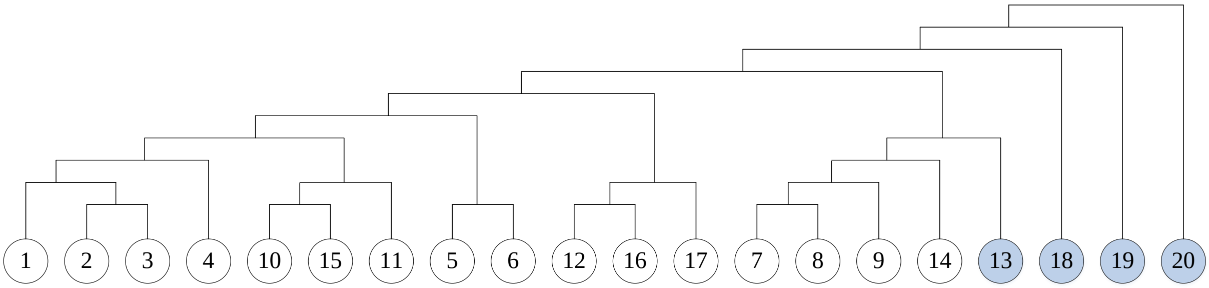

Figure. 2 Clustering results of UPGMA algorithm

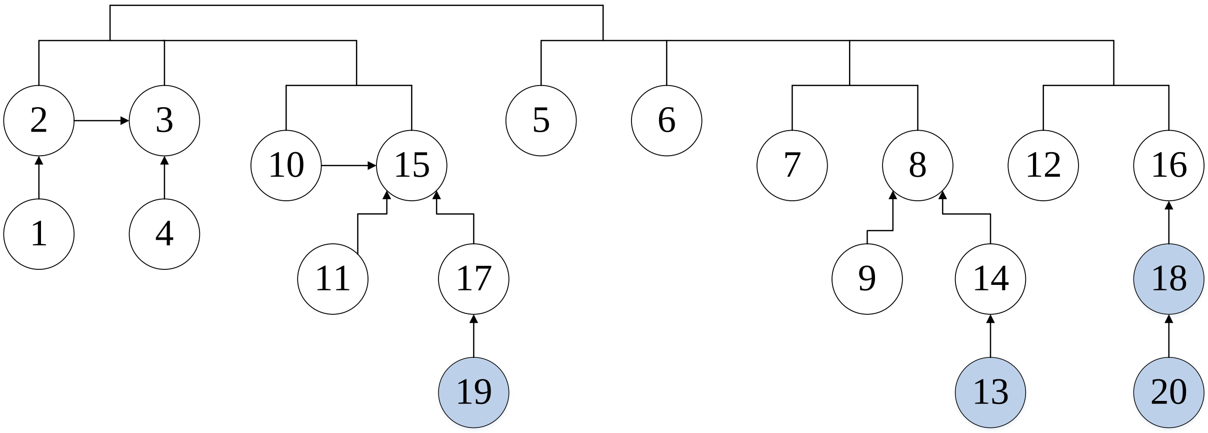

Figure. 3 Clustering results of RS algorithm

The comparative experiments show that the clustering results of the RS algorithm on most of the test data sets are better than the two classical algorithms of UPGMA and K-means. It is worth mentioning that RS algorithm is a hierarchical clustering algorithm but has linear time complexity. This makes it possible to improve the problem of low efficiency without losing the stability of traditional hierarchical clustering algorithm. RS algorithm is widely used in the future because of its excellent efficiency.

4. Research on the detection algorithm of merchant category code application

4.1 Model design

Through comparison, it is found that the transaction amount of merchants is also regular, and the rule based on transaction amount of each merchant type and the rule based on trading volume are basically consistent. Figures 4, 5 and 6 show the comparison of day-based trading volume and transaction amount in large warehouse supermarkets (a), airlines (b), dining places and restaurants (c). It can be concluded from the figure that the rule of trading volume and transaction amount is basically consistent. Therefore, only the trading volume is used to carry out work in subsequent studies.

Therefore, it is verified that merchants in different industries do have their business rules. The business rule of individual merchant, namely the rule of trading volume based on day, week and month, is called the “behavior pattern” of the merchant. And the overall business rule of a certain industry is called the “industry model” of this industry.

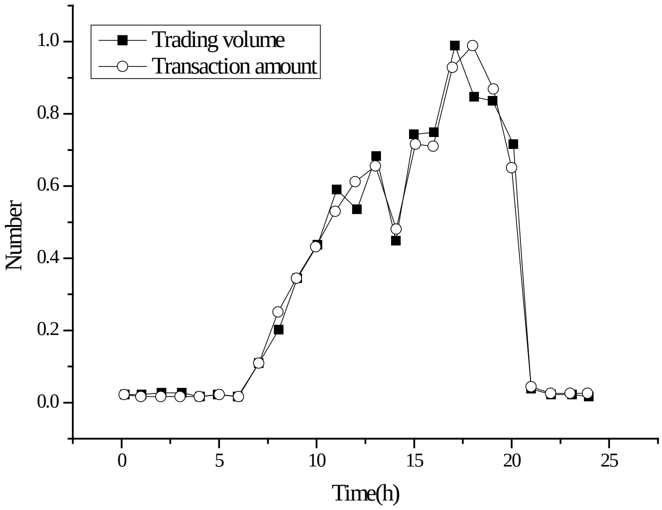

Figure. 4 Comparison of daily trading volume and transaction amount in a large warehouse supermarket

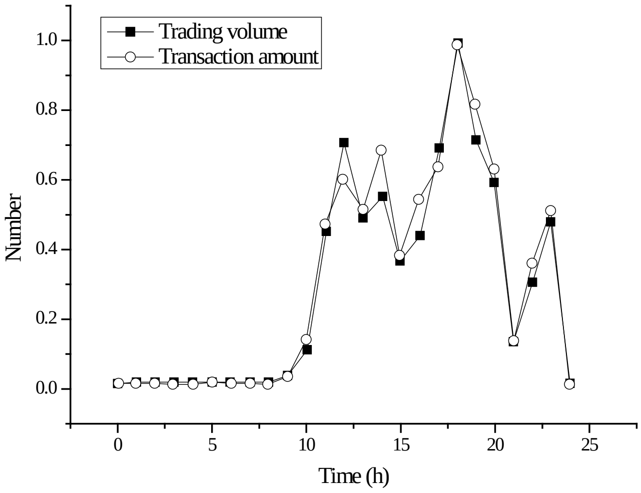

Figure. 5 Comparison of airline’s day-based trading volume and transaction amount

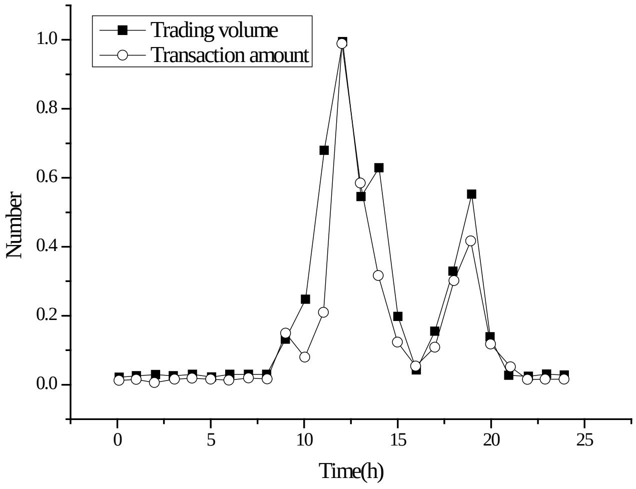

Figure. 6 Comparison of daily trading volume and transaction volume of dining places and restaurants

The industry model represents the overall business law of the whole industry, while the merchant's behavior model describes the business situation of the individual merchant. For normal merchants that don't apply MCC, their behavior pattern should be the same or similar to the industry pattern of the MCC. On the other hand, for MCC application merchants, their behavior pattern should be the same or similar to the industry model of the MCC where the real industry is located. At the same time, there should be differences in the industry model of application MCC. There are differences between the industry model and the MCC application merchant behavior model. The MCC application merchant, that is, the MCC of the business that the merchant actually operates is different from the MCC it applies for. Then the behavior model of the MCC application merchant should be consistent with the industry model of the real industry and is different from the industry model of the MCC it applies for. How big the difference is and whether it is enough to identify the application merchants will need to be validated with data. Figure 7 shows the MCC application legend for an abnormal merchant. The three curves in the figure represent the day-based behavior pattern of an abnormal merchant of the department store, the day-based industry model of the dance hall, and the day-based industry model of the department store.

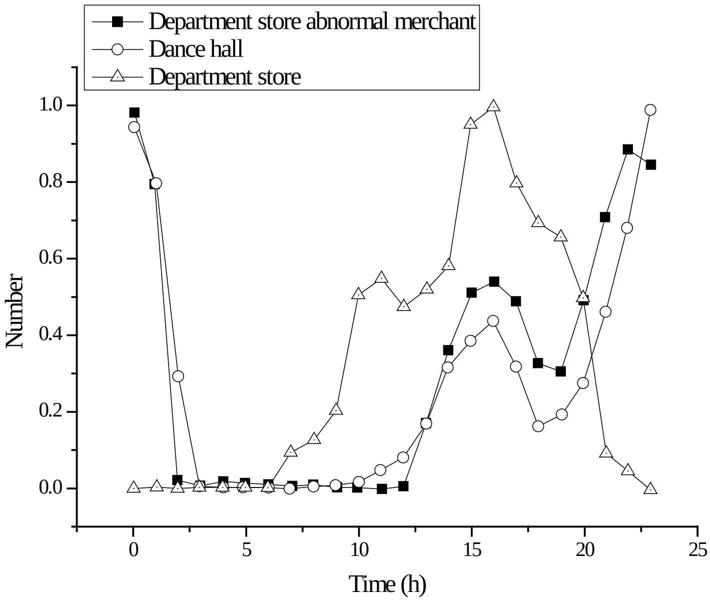

Figure. 7 MCC application merchant legend in department store

It can be concluded from the figure that the industry model of the department store should be twin peaks, with the peak appearing around 4 p.m. The behavior patterns of the abnormal department store merchants in the picture are completely different from those of the department store industry model, but they are very similar to the industry model of the dance hall. The peak of transactions occurred during the time when the department store had stopped operating, around midnight. Comparing the rates of department stores and dance halls, it is found that the rates of dance halls are higher than those of department stores. This indicates that this merchant is likely to be the MCC application merchant.

Through the verification of the above two problems, it is considered that it is feasible to compare the industry model and the merchant behavior pattern and observe the difference to judge whether the merchant applies the MCC. Next, it discusses how to identify application by comparing industry models and merchant behavior patterns.

In fact, this is the problem of dichotomizing merchants. For a merchant, it has only two identities: MCC application merchants and normal merchants without MCC application. If the dichotomy is used to solve the problem of application identification, the first problem to be solved is how to select the appropriate feature to form the feature vector. For the case of sufficient data, the behavior pattern of the merchant is directly used as the feature vector, and the binary classification modeling of the merchant under the same MCC can complete the detection of the MCC application merchant. It needs to consider how to model from a holistic perspective. The previous article has verified the difference between the industry modes and the difference between the industry mode and the MCC application merchant behavior mode. These two big differences are the key to the overall data modeling. By calculating and extending the inter-industry and intra-industry differences, a series of parameters can be obtained, which are suitable to be used as feature vectors for classification modeling. After the feature selection is completed, the classification model is trained with the feature vector to generate the classifier to identify whether the merchant has MCC application behavior. The trained classifier can be used to judge whether the merchant has application behavior. Figure 8 shows the merchant category code application detection model. The model is mainly composed of feature selection module and classification module. The feature selection module includes three sub-modules: data preprocessing, industry mode database training, and feature transformation. The three sub-modules are described in detail in the next section.

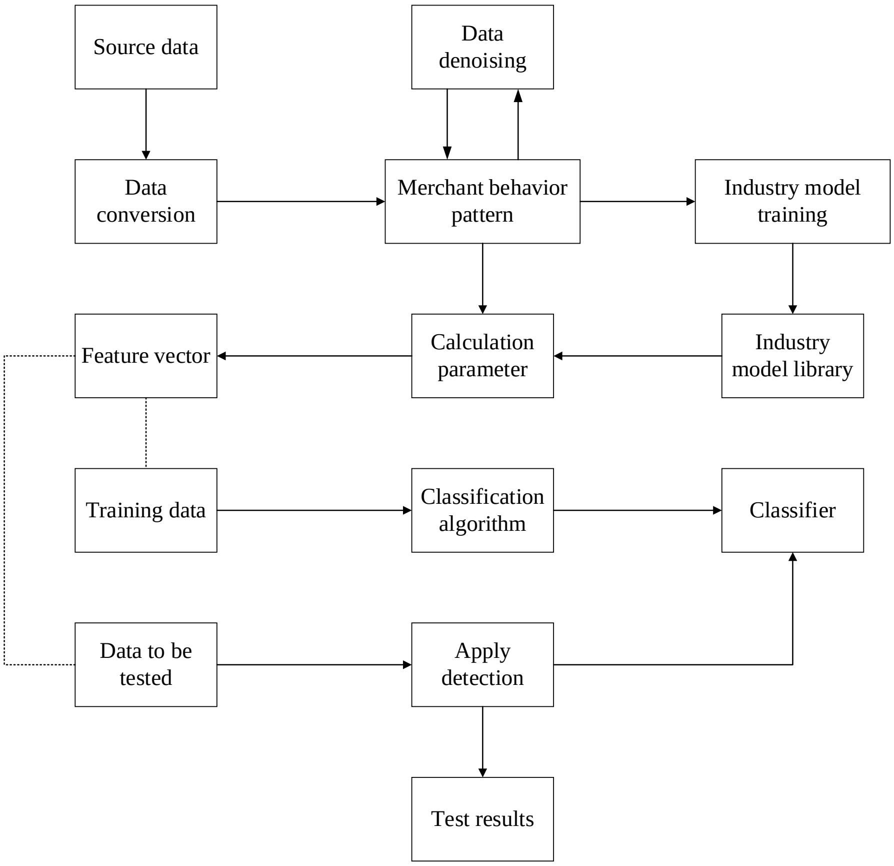

Figure. 8 Merchant category code application detection model

4.2 Feature selection module

The feature selection module is the core part of the merchant class application detection model. In the feature selection module, there are mainly three steps:

Step 1: data pre-processing. Carry out statistics, calculation and denoising on the source data to generate the behavior pattern data of merchants with strong readability; step 2: industry model library training. The clustering algorithm is used to train the industry models of various industries and organize them together to form the industry model database; step 3: feature transformation. The industry pattern library is used to calculate the difference between the merchant behavior pattern and each industry, generate the features to identify the MCC application merchant, and organize them into feature vectors.

4.3 Experiment and analysis

After the feature transformation generates the feature vector, a key step can be taken: these feature vectors are used to train the classifier, and the trained classifier is used to identify the merchant MCC.

Data description:

The basic information of the data sets used in the experiment is shown in table 2. The labels in these data sets are manually labeled by the Unionpay operating department. However, there is a problem in the data that a small amount of mark time doesn't match the time of the merchant data. That is, the marked time for a merchant to be an MCC merchant is January 2017, but the time for obtaining these data is the whole year of 2015. Thus, the merchant that has been corrected is still marked as MCC application merchant and use the uncorrected MCC.

Table 2 Basic information of experimental data

DataSet	Number of MCC-applied merchants	Number of normal merchants	The total number of merchants
I	5415	7969	13384
II	5428	6951	12379
III	601	888	1489
IV	1413	5086	6499

Evaluation index:

The data sets of merchant behavior used in the experiment all have classification labels, so the algorithms are evaluated with Precision, Recall, F1 and time consumption. Precision refers to the proportion of merchants in which the actual label is actually applied in the merchant whose predict label is applied. Table 3-2 is the relationship table between actual label and predicted data: TP is True Positive; FP is False Positive; FN is False Negative; TN is Ture Negative. The calculation method of Precision is shown in formula (7):

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

Recall rate refers the percentage of merchants whose label is predicted to be applied by the algorithm in all merchant whose actual label is applied. The calculation method of recall rate is shown in formula (8):

$$\mathrm{Re}\,call = \frac{TP}{TP + FN} \tag{8}$$

F1 is a commonly used evaluation index in information retrieval, which is used to comprehensively evaluate the precision and recall rate of the algorithm. The calculation method of F1 is shown in formula (9):

$$F1 = \frac{2 \times \mathrm{Precision} \times \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}} \tag{9}$$

Time consumption is calculated by the difference between the start time and the end time of the algorithm.

4.4 Experimental result

The performance of each algorithm on the data set is tested with 10 cross validation. That is, the data set is randomly divided into 10 equal parts, and one part is taken as the test set at a time, while the other nine are merged as the training set. Table 3 shows the results of 10 cross validation tests of dataset I. The results show that naive Bayes algorithm takes the least time, while C4.5 has the optimal evaluation index.

Table 3 The results of 10 cross validation tests of dataset I

Algorithm name	Accuracy rate	Recall rate	F1	Time consumption (seconds)
Naive Bayes	0.832	0.900	0.865	0.004
Logistic regression	0.867	0.877	0.872	0.40
RBF artificial neural network	0.824	0.893	0.851	0.98
Decision tree C4.5	0.907	0.872	0.889	0.43
Random forest	0.886	0.882	0.884	1.15

Table 4 shows the results of 10 cross validation tests of dataset II. It can be concluded from the results that the least time-consuming algorithm is the Naive Bayes algorithm, and its recall rate is better than other algorithms. And C4.5 has the optimal accuracy rate. Although the accuracy rate and recall rate of random forests are not the highest, they are all at a high level, so the random forest has the highest F1 value.

Table 4 The results of 10 cross validation tests of dataset II

Algorithm name	Accuracy rate	Recall rate	F1	Time consumption (seconds)
Naive Bayes	0.833	0.901	0.866	0.16
Logistic regression	0.869	0.879	0.874	0.56
RBF artificial neural network	0.832	0.889	0.859	0.86
Decision tree C4.5	0.901	0.866	0.883	0.56
Random forest	0.886	0.882	0.884	1.18

Table 5 The results of 10 cross validation tests of dataset III

Algorithm name	Accuracy rate	Recall rate	F1	Time consumption (seconds)
Naive Bayes	0.844	0.914	0.878	0.01
Logistic regression	0.875	0.883	0.879	0.05
RBF artificial neural network	0.854	0.884	0.869	0.08
Decision tree C4.5	0.871	0.883	0.877	0.03
Random forest	0.867	0.881	0.874	0.09

Table 5 shows the results of 10 cross validation tests of dataset III. Dataset III has a smaller amount of data than other dataset. It can be concluded that naive Bayes algorithm still has the best recall rate. On the whole, logical regression is better.

The data distribution in each data set is different. And since the labels in these data sets are manually marked by the business department of Unionpay, there is a small amount of mismatch between marking time and merchant data time in the data. The so-called “actual label” is not necessarily true. The error caused by this label has a certain influence on the experimental results. This influence varies with the data that Unionpay provides and is unavoidable in non-production environments. Table 6 shows the results of 10 cross validation tests of dataset IV.

Table 6 The results of 10 cross validation tests of dataset IV

Algorithm name	Accuracy rate	Recall rate	F1	Time consumption (seconds)
Naive Bayes	0.374	0.483	0.422	0.03
Logistic regression	0.508	0.053	0.096	0.17
RBF artificial neural network	0.000	0.000	0.000	0.27
Decision tree C4.5	0.556	0.193	0.286	0.16
Random forest	0.452	0.399	0.424	0.60

It can be concluded that the test results of the dataset VI declined as a whole. But the naive Bayes algorithm still has better performance than other algorithms. The reasons for the decline in overall effect are analyzed, including the following two aspects: the MCC distribution in the data set is too centralized; the error rate of labels in data sets is too high. From the point of view of selected feature vectors, most of them are based on relative quantities with other MCC. For example, the expense grade and expense grade difference of the most similar MCC are based on the rate comparison with the most similar MCC. When the training set is too concentrated on a certain MCC, these features based on relative quantity make it difficult to distinguish the difference between the applied merchant and the non-applied merchant.

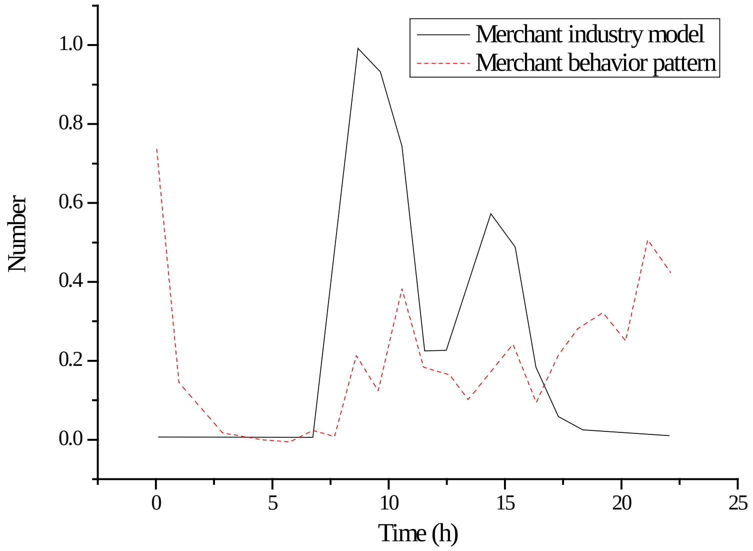

Figure. 9 The comparison between the merchant behavior pattern and the industry pattern of abnormal merchants based on the day

Based on the whole experiment, the current detection algorithm of merchant category code application is indeed feasible. The accuracy rate and recall rate of the detection can reach more than 80% (except the special data set IV). By comparing the experimental results of each classification algorithm, the classifier used in the subsequent work should take into account naive Bayes and logistic regression algorithms.

5. Conclusion

The merchant category code application behavior disrupts business rules and causes a significant economic loss to the Unionpay institutions. To reduce the labor cost and recover the economic loss brought by merchant MCC application to Unionpay, a set of merchant category code application detection model is designed. In this model, data pre-processing is first performed on the source data: the scattered merchant transaction data is used to obtain statistical values of the trading volume of these merchants in days, weeks and months, and calculate the first derivative of these statistical values. These 6 types of data are taken as the merchant behavior pattern, and the noise in the data of merchant behavior pattern is removed with clustering algorithm. Then train the industry model: the merchant behavior patterns under each MCC are trained with clustering algorithm for the industry patterns of each MCC, and the industry patterns of each MCC are integrated together to form an industry pattern library. Then, combined with the merchant's behavior pattern and the industry model of each MCC, the feature vectors required by the classifier are generated through a series of calculations. Finally, these feature vectors can be used to train the classifier to detect the data that needs to be detected.

A series of experiments show that the design of the merchant category code application detection model is feasible. Both the accuracy rate and recall rate of the application detection can reach a satisfactory value. However, there is still much room for improvement in the algorithm based on the experimental results. The focus should be on the core of feature selection. It is believed that after further optimization and adjustment, the model will be applied to the production data of Unionpay, which will greatly reduce the labor cost and recover the economic loss caused by the application of merchant category code.

Acknowledgements

The authors acknowledge the National Natural Science Foundation of China (Grant: 111578109), the National Natural Science Foundation of China (Grant: 11111121005).

References

[1] Alex G.C. de Sá, Pereira A C M, Pappa G L. A customized classification algorithm for credit card fraud detection [J]. Engineering Applications of Artificial Intelligence, 2018, 72: 21-29.

[2] Carneiro N, Gonçalo Figueira, Costa M. A data mining based system for credit-card fraud detection in e-tail[J]. Decision Support Systems, 2017, 95.

[3] Dal A P, Boracchi G, Caelen O, et al. Credit Card Fraud Detection: A Realistic Modeling and a Novel Learning Strategy [J]. IEEE Transactions on Neural Networks & Learning Systems, 2018, 29(8): 3784-3797.

[4] Patil S, Nemade V, Soni P K. Predictive Modelling For Credit Card Fraud Detection Using Data Analytics[J]. Procedia Computer Science, 2018, 132: 385-395.

[5] Bahnsen A C, Aouada D, Stojanovic A. Feature engineering strategies for credit card fraud detection[J]. Expert Systems with Applications An International Journal, 2016, 51(C): 134-142.

[6] Adewumi A O, Akinyelu A A. A survey of machine-learning and nature-inspired based credit card fraud detection techniques[J]. International Journal of System Assurance Engineering & Management, 2017, 8(S2): 1-17.

[7] Van Hardeveld G, Webber C, O'Hara K. Discovering credit card fraud methods in online tutorials[J]. 2016.

[8] LafondLapalme, Joël, Duceppe M O, Wang S, et al. A new method for decontamination of de novo transcriptomes using a hierarchical clustering algorithm[J]. Bioinformatics, 2016, 33(9).

[9] Chattopadhyay M, Sengupta S, Sahay B S. Visual hierarchical clustering of supply chain using growing hierarchical self-organising map algorithm[J]. International Journal of Production Research, 2016, 54(9): 1-20.

[10] Siless V, Chang K, Fischl B, et al. AnatomiCuts: Hierarchical clustering of tractography streamlines based on anatomical similarity[J]. Neuroimage, 2016, 166: 32-45.

[11] Neamatollahi P, Naghibzadeh M. Distributed unequal clustering algorithm in large-scale wireless sensor networks using fuzzy logic[J]. Journal of Supercomputing, 2018: 1-24.