

# Algorithm of Non-Redundant Association Rules based on User Interest Orientation

Xiaodong Qian<sup>a,\*</sup> and Rui Guo<sup>b</sup>

<sup>a</sup>*School of Economics and Management, Lanzhou Jiaotong University, Lanzhou, 730070, China*

<sup>b</sup>*School of Electronics and Information Engineering, Lanzhou Jiaotong University, Lanzhou, 730070, China*

---

## Abstract

Traditional association rule mining algorithms usually generate association rules that satisfy the predefined minimum support and confidence threshold. However, with the rapid expansion of the data scale, the set of association rules obtained from traditional association rules contains massive redundant rules that cannot reflect the specific interests of users. In this paper, we combine the subjective interestingness optimization algorithm with the objective redundant association rule deletion algorithm in an effort to complement each of their shortcomings and improve optimization ability. We also present an algorithm for non-redundant association rules based on user interest orientation. First, an algorithm is proposed to remove redundant association rules based on the first-order predicate formula. Then, the non-redundant association rule set is used as the original data set for classification. By using the attributes or rules of the user's interest as the guiding object, the template idea is added as the carrier of the user's meaning. Based on the user's prior knowledge, the template is divided into three types: GI, RPC, and PK. Based on this method, the association rules are obtained using the method of interest calculation. The experimental results show that the algorithm has been improved effectively.

*Keywords:* association rules; user interest; first-order predicate

(Submitted on May 9, 2018; Revised on June 16, 2018; Accepted on July 23, 2018)

© 2018 Totem Publisher, Inc. All rights reserved.

---

## 1. Introduction

Association rule mining is an important branch of data mining, and it can be used to find interesting relations among items and attributes of databases. These relations are often previously unknown and hidden, meaning they cannot be obtained through logic operations or statistical methods of the database. The association rule mining algorithm usually generates rules that require all data to meet the predefined minimum support degree and confidence threshold [1], but due to the rapid expansion of data that is collected and stored, association rule sets containing massive redundant rules are obtained. As a result, the interpretations of rules are destroyed because many association rules are generated. Users cannot achieve a direct understanding and analysis of the rules through the effective method. The association rules found in the support degree-confidence framework reflect the relationship between the current data items, which cannot accurately reflect the real relationship between the attributes of the data set. The traditional method also produced false, redundant, and conflicting rules and cannot ensure that rules are established in all data. Association rules mining focuses on discovering how association rules gradually shift and how to select effective rules from a large number of association rules, so as to provide better services for users.

Selecting and assessing numerous association rules is a problem faced by many users and experts today. A valuable association rule must be easy to understand, easy to operate, and can be tested to reflect association rules of different data. Interest [2] degree is an important branch of association rule mining and even data mining as a whole, but there are few data mining systems to solve this problem. Piatetsky-Shapiro first proposed the concept of interest [3]. Interest is a relative concept; for different areas and experts, the rules of interest are different, and even the same area of interest may differ for different time requirements. To evaluate the interest degree of rules, researchers have put forward many methods to measure the interest degree. Current methods can be divided into the objective method and the subjective method.

\* Corresponding author.

E-mail address: [qianxd@mail.lzjtu.cn](mailto:qianxd@mail.lzjtu.cn)

The objective interest measure method is mainly based on probability theory [4], statistic theory, and information theory, and the method depends on the form of rules and basic data during the mining process. Interest rules reflect the statistical relationship between statistical variables. The support degree, integrity degree, and confidence degree are always the main factors of the objective interest degree, and the measurement criterion of different interest degrees is formed by transformation and deformation. Different interest degree measurement methods calculate the value of the interest degree from different angles. The greater the value of interest degree, the greater the value of the corresponding rules. The association rules with higher degrees of interest are used as the rules of interest, solving the problem of evaluating the interest degree to a certain extent.

At present, the objective interest degree is common in association rules mining, but the objective interest degree by itself is not enough. It is difficult to consider patterns and rules in all directions. Through the comparison of different methods of the interest measurement, it is found that most of the measurement methods are not consistent with the calculation results of interest rules and are sometimes even contradictory. Simulation results of various methods in different field data show that different interest degree measure methods for generating rules are not consistent when ranking according to the interest degree. Currently, there is not an optimization objective interest degree measure method that can deal with various data directly and effectively. In essence, the issue of interest degree is subjective and requires the user to support it with knowledge and experience in the field of expertise.

In terms of subjective interest, in the context of field independence [5], Avisilberschatz divided the subjective rules into two kinds: unexpected and actionable. Unexpected rule is the rule of conflict with expectations, while actionable rule can provide a reference for users to do their own things. The subjective interest method from another perspective is to analyse association rules for the purpose of finding out the rules that the user is interested in. In addition, a template is used as a tool to specify the domain knowledge information, which can be used to analyse the subjective interest. Hoschka and Klosgen were the first to propose the concept of templates. The deviation analysis method measures the distance between the actual and expected results, and Piatetsky-Shapiro and Matheus combined the deviation and interest degree to analyse the interest degree of deviation. Klemettinen et al. proposed the concept of Rule Template and used the including template and the restriction template to filter the interesting rules and non-interesting rules, respectively. In addition, a number of similar rule templates have been proposed in order to describe the rules that are expected by users.

In terms of user needs, the research of interest degree has the following two significances:

- 1) In the face of mining rules, the only goal of the user is to find special rules that have not yet been found. If we only provide out of order rules, users will need to spend more time to analyse and discover valuable rules.
- 2) Users want to quickly cut into the theme and find valuable information when there are thousands of rules. Therefore, it is mainly aimed at the optimization of interest and proposes a more perfect optimization algorithm.

The calculation of the interest degree method is undeserved and the users' meaning is limited. These are the main problems in the subjective interest degree optimization algorithm. By combining the subjective interest degree optimization algorithm with the deletion algorithm of objective redundant association rules, optimization ability can be improved. Firstly, the non-redundant association rules are obtained using the objective redundant association rule algorithm. Then, the generated association rules are classified. By taking the attributes or rules of the user's interest as a guide, the template idea is added as a carrier to express the users' meaning and classify the template. The measuring method of interest degree is optimized to improve the calculation method of the interest degree.

## 2. Basic Concepts

### 2.1. Degree of Interest

The degree of interest can be defined in a variety of ways, but it is related to confidence and support for any measure of interest. It should at least meet the three basic requirements proposed by Shapiro-Piatetsky.

- 1) When the confidence of a rule is equal to the correct rate of the background knowledge, the degree of interest is zero.
- 2) When the support degree (confidence) is certain, the degree of interest increases monotonically with the degree of confidence (support).
- 3) When the number of the correct response is a certain number, the degree of interest decreases monotonically with the degree of support.

## 2.2. User Knowledge Classification

If users want to express a certain area of knowledge, they must provide a set of rules for the representation of knowledge. These rules are called the user knowledge description language.

Users' knowledge can be divided into three types: general impression of knowledge, relatively accurate concept, and precise knowledge, respectively called GI, RPC, and PK. A user knowledge description language also includes three types of knowledge description. In the above classification, the first two categories represent the user's fuzzy concepts, and the latter represents the precise knowledge of the user.

### 2.2.1. Knowledge Description of GI

GI knowledge shows the user's fuzzy concept to association rules, and users believe that there is a certain relationship between some items or categories. However, it cannot determine the association and is expressed as Equation (1):

$$gi(<S_1, S_2, \dots, S_m>) [sup, conf] \quad (m > 0, m \text{ belong to integer}) \quad (1)$$

The above explanation is as follows: Using  $A_i \in A$  to represent an item, if  $<a_1, \dots, a_m>$  and  $<b_1, \dots, b_n>$  are an instance of  $<S_1, S_2, \dots, S_m>$ , then the association rules  $a_1, \dots, a_m \rightarrow b_1, \dots, b_n$  are consistent with GI. Otherwise, it is unexpected for GI.  $sup$  and  $conf$  are chosen arbitrarily.

GI knowledge is actually an or-expression, and each item of this expression represents a general impression of the user. For example, in Equations (2) and (3),  $gi <a, b, c>$  is extended to the "or" expression:

$$((a \wedge b) \rightarrow c) \vee ((a \wedge c) \rightarrow b) \vee ((c \wedge b) \rightarrow a) \vee (c \rightarrow (a \wedge b)) \quad (2)$$

$$\vee (c \rightarrow (a \wedge b)) \vee (a \rightarrow (c \wedge b)) \vee (b \rightarrow (a \wedge c)) \quad (3)$$

### 2.2.2. Knowledge Description of RPC

RPC knowledge shows that users have a more accurate concept of association rules, and they believe that there is a correlation between the categories or items and the direction of the association. This can be expressed in the following form:

$$pk(<S_1, S_2, \dots, S_m \rightarrow V_1, V_2, \dots, V_n>) [sup, conf] \quad (m, n > 0, m, n \text{ belong to integer}) \quad (4)$$

The explanation of Equation (4) is as follows:

Using  $S_i$  and  $V_i$  to respectively represent an item, if  $a_1, \dots, a_m \rightarrow b_1, \dots, b_n$  is an instance of  $S_1, S_2, \dots, S_m \rightarrow V_1, V_2, \dots, V_n$ , the association rules are consistent. Otherwise, it is unexpected for RPC.  $sup$  and  $conf$  are chosen arbitrarily.

### 2.2.3. Knowledge Description of PK

PK knowledge shows that users have an accurate concept of association rules, and they believe that there is a certain relationship between the class or items and the direction of the association according to existing knowledge of the support degree ( $sup$ ) and confidence. It can be expressed in the following form:

The explanation of the Equation (4) is also as follows:

Using  $S_i$  and  $V_i$  to respectively represent an item, if  $a_1, \dots, a_m \rightarrow b_1, \dots, b_n$  is an instance of  $S_1, S_2, \dots, S_m \rightarrow V_1, V_2, \dots, V_n$ , the main body of the association rules  $a_1, \dots, a_m \rightarrow b_1, \dots, b_n$  is consistent with PK, and whether there is an accident depends on the comparison of the rules and instructions of the support and confidence.  $sup$  and  $conf$  must be selected.

## 2.3. Classification of Association Rules

Refer to  $U$  as the user knowledge space and  $A$  as a set of association rules. The rules of  $A$  can be divided into four categories: the consistency rule, result of unexpected rules, conditional unexpected rules, and the unexpected rules of the two.

(1) The consistency rule: If part of the condition and results of the rule  $A_i \in A$  match with a knowledge description of users  $U_j \in U$ , then this is called the consistency rule, and the consistency of the matching degree is recorded as  $Conf_{ij}$ .

(2) The result of unexpected rules: If part of the condition of the rule  $A_i \in A$  matches with a knowledge description of users  $U_j \in U$ , but the part of the results was not well matched, then this is called the result of unexpected rules. The matching degree of the result of unexpected is recorded as  $unexpConseq_{ij}$ .

(3) Conditional unexpected rules: If part of the results of rule  $A_i \in A$  matches with a knowledge description of users  $U_j \in U$ , but part of the condition was not well matched, then this is called conditional unexpected rules. The matching degree of the conditional unexpected is recorded as  $unexpCond_{ij}$ .

(4) The two-unexpected rules: If part of the condition and results of the rule  $A_i \in A$  have a worse match with a knowledge description of users  $U_j \in U$ , then this is called the two-unexpected rules. The matching degree of the two-unexpected is recorded as  $bsUnexp_{ij}$ .

### 3. The Introduction of the Algorithm

The main idea of the algorithm is to first obtain the association rules from the original data by using Weka.

Firstly, we get the association rules from the original data by using Weka. Secondly, we remove the redundant association rules algorithm by using first order predicate formula, and then we delete the association rules that we have obtained. Thirdly, we take the non-redundant association rule set as the original data set and classify it by association rules. We change the user's prior knowledge into GI, RPC, and PK. Finally, we use the calculation method of interest degree to get the association rules.

#### 3.1. Using Weka to Generate Association Rules

Weka is an open source software based on Java platform. It can make many operations such as the pre-processing of data, classification, clustering, association rules, regression, and visualization. Because the code is open source, it can be convenient to modify and test arithmetically.

Association rules are generated from the original data by Weka in this paper, as shown in Figure 1. The specific steps are as follows:

**Step 1** Load raw data.

**Step 2** Select the corresponding association rules arithmetic and set the parameters.

**Step 3** Execute the arithmetic to obtain the result.

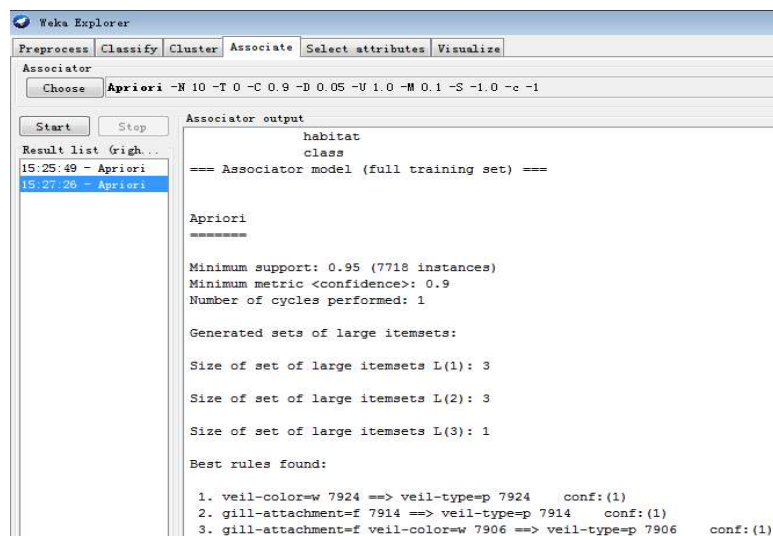


Figure 1. Running the Apriori algorithm in Weka

### 3.2. The Algorithm of First Order Predicate Formula to Remove the Redundant Rule

The algorithmic idea of the first order predicate formula to remove the redundant rule algorithm is: we use the first order predicate formula to represent the association rules and transform it through the equivalent formula; then, using the equivalence between algorithm and matrix, the predicate formula is converted to adjacency matrix and is deleted by the redundant rule algorithm.

The arithmetic of transforming the first order predicate formula into the correlation matrix is as follows:  $P \rightarrow Q$ .

The paper uses the first order predicate formula to represent the association rules, using the implication of predicate calculus, called:  $P \rightarrow Q \Leftrightarrow \neg(P \wedge \neg Q)$ .

#### 3.2.1. The Arithmetic of Transforming the First Order Predicate Formula into the Correlation Matrix.

**Arithmetic 1:** Assume that  $P$  is a paradigm only containing  $\neg$  or  $\wedge$ ,  $I$  is a matrix variable of the number of ranks. The first column of  $I$  is the ordinal list [6].

**Step 1**  $I = (P, 1)$ , this is the initial value, with 1 rows and 2 columns.

**Step 2** If the sequence of  $I$  contains  $\neg$  or  $\wedge$ , then we take this sequence as  $X$ .

① If  $X$  is equal to  $\neg Q$ ,  $X$  will exchange to  $Q$  and multiply the value of the other node by -1.

② If  $X = Q_1 \wedge X = Q_2$ , we add two new lines which are marked as sequence  $Q_1$  and  $Q_2$  in  $I$  and use D as the initial value of the two elements. If  $X$  row  $j$  column of the node value is -1, the value of the  $Q_1$  and  $Q_2$  row  $i$  column 0 exchange to -1. The value of the  $X$  row  $i$  column -1 exchange to 0; if  $X$  row  $j$  column of the node value is 1, the  $j$  column of the node value 1 exchange to 0, the  $j$  column will become into two columns including column  $Q_1$  0 exchanging to 1 and another column  $Q_2$  exchanging 0 to 1.

Repeat the above operation until the  $X$  line node value is 0, and delete this line.

Repeat Step 2 until the sequence column does not contain  $\neg$  or  $\wedge$ .

**Step 3** the sequence column of  $I$  is  $B$ . If there are two different lines containing the same sequence, we seek the value of each column in the two rows; if the line is -1 and the other row is 1, we refer to it as a tautology. We know that if  $P$  is also a tautology, then the arithmetic ends. If there is no such column, we copy the nonzero elements of a near row in the next line which exchanges the front column 0 to the same column nonzero value and then deletes the previous column.

Repeating the sequence of the same row of the above statement, we continue the process until there are no identical sequences in the sequence.

Aiming at the B element that does not appear in the sequence, the row's remaining value is set to 0, and the algorithm ends. In the generated association matrix, 1 represents the smooth and -1 represents the inverse.

Aiming at the B element which does not appear in the sequence column, we add a row by its sequence and the rest of this row is 0, ending the arithmetic. This generates the associated matrix, where 1 represents the forward direction and -1 represents the inverse of the reverse direction.

#### 3.2.2. Correlation Matrix is Transformed to Adjacency Matrix

The correlation matrix is the relationship between each point and the side that is represented by a matrix determinant. The adjacency matrix is a matrix that represents the relationship between the point and the point of a matrix determinant. They are both the matrix representation of the different forms of association rules and can be converted to each other. We can achieve the specific algorithm through MATLAB and the use of references [7].

### 3.2.3. The Algorithmic of Removal of Slave Rule

1) For each column of the adjacency matrix, use the back-rules of association rules. To find the column for 1, the line corresponding to the position of 1 means the back-rules of association rules and has a meet operation. If the results have a meet, the corresponding line containing a number of frequent items is exchanged for 0. In the same way, each line is processed as above [8].

2) To traverse the entire matrix, if a row or column values all equal 0, delete the row and column.

3) To traverse the end of the row and column of the matrix, if the row or column all equal 0, delete the row or column until a column or row equalling 1 is found.

4) The rest of the matrix is the adjacency matrix after removing the slave association rules, and the existence of 1 node is the residual association rule.

5) We remove the residual adjacency matrix after the removal of the subordinate association rules from the association rule.

### 3.3. Calculation for Degree of Interest Calculation

A rule has two parts, the condition and the result. We need to compute the matching degree between them and the user knowledge. Use  $L_{ij}$  and  $R_{ij}$  to represent the conditions and results of the rule  $A_i$ . Use  $L_{ij}$  and  $R_{ij}$  to represent the matching degree between the condition and result of the rule  $A_i$  and a user knowledge description  $U_i$ . The matching degree is expressed by the percentage of the number of matches. Its calculation method is as follows:

Suppose the rules  $A_i$  that are investigated are  $a_1, \dots, a_m \rightarrow b_1, \dots, b_n$ .

(1) When  $U_j$  is a description of GI,  $gi(<S_1, S_2, \dots, S_m>) [sup, conf]$ ,  $L_{ij}$  and  $R_{ij}$  are respectively calculated by Equations (5) and (6).

$$L_{ij} = \frac{LM_{ij}}{LN_i} a + \frac{SM_{ij}}{SN_i} (1-a) \quad (5)$$

$$R_{ij} = \frac{RM_{ij}}{RN_i} a + \frac{SN_{ij}}{SN_i} (1-a) \quad (6)$$

(2) When  $U_j$  is a description of RPC,  $rpc(<S_1, S_2, \dots, S_m \rightarrow V_1, V_2, \dots, V_n>) [sup, conf]$ ,  $L_{ij}$  and  $R_{ij}$  are respectively calculated by Equations (7) and (8).

$$L_{ij} = \frac{LM_{ij}}{LN_i} a + \frac{LSM_{ij}}{LSN_i} (1-a) \quad (7)$$

$$R_{ij} = \frac{RM_{ij}}{RN_i} a + \frac{SVM_{ij}}{SVN_i} (1-a) \quad (8)$$

(3) When  $U_j$  is a description of PK,  $pk(<S_1, S_2, \dots, S_m \rightarrow V_1, V_2, \dots, V_n>) [sup, conf]$ , the values of the  $unexpConseq_{ij}$ ,  $unexpCond_{ij}$  and  $bsUnexp_{ij}$  have no practical significance. We only use  $Conf_{ij}$  to measure the degree of subjective interest of the rule, and its calculation is shown in Equation (9).

If the subjects match between the rules and the description:

$$confm_{ij} = 1 - \frac{|RS - SS|}{\max(RS, SS)}a + \frac{|RC - SC|}{\max(RC, SC)}(1 - a) \quad (9)$$

ELSE

$$Conf m_{ij} = 0$$

The above meanings are as follows:

$LM_{ij}$ : The number of matches between the condition part of  $A_i$  and the GI,  $RM_{ij}$ : The number of matches between the results part of  $A_i$  and the GI,  $SM_{ij}$ : The number of matches from GI to  $A_i$ ,  $LN_i$ : The number of the condition part of  $A_i$ ,  $RN_i$ : The number of the result part of  $A_i$ ,  $SN_i$ : The total number of GI,  $LSM_{ij}$ : The number of matches between the condition part of  $A_i$  and the RPC,  $RVM_{ij}$ : The number of matches between the result part of  $A_i$  and the RPC,  $LSN_i$ : The number of the condition part of RPC,  $RVN_j$ : The number of the result part of RPC,  $RS$ : The support of rule  $A_i$ ,  $SS$ : The support of description PK,  $RC$ : The confidence of rule  $A_i$ ,  $SC$ : The confidence of description PK.

(4) The calculations of  $unexpConseq_{ij}$ ,  $unexpCond_{ij}$ ,  $bsUnexp_{ij}$  and  $Conf m_{ij}$  are shown in Equations (10), (11), (12), and (13).

$$Conf m_{ij} = L_{ij} \times R_{ij} \quad (10)$$

$$unexpConseq_{ij} = \begin{cases} 0, & L_{ij} - R_{ij} \leq 0 \\ R_{ij} - L_{ij}, & L_{ij} - R_{ij} > 0 \end{cases} \quad (11)$$

$$unexpCond_{ij} = \begin{cases} 0, & R_{ij} - L_{ij} \leq 0 \\ R_{ij} - L_{ij}, & R_{ij} - L_{ij} > 0 \end{cases} \quad (12)$$

$$bsUnexp_{ij} = 1 - \max(confm_{ij}, unexpConseq_{ij}, unexpCond_{ij}) \quad (13)$$

We use  $L_{ij} - R_{ij}$  to calculate  $unexpConseq_{ij}$ , because we hope that the results of Unexpected Rules have a high degree of matching in the condition part and that the results of the matching degree are low. This is the same reason that we use  $R_{ij} - L_{ij}$  to calculate  $unexpCond_{ij}$ . The calculation formula of  $bsUnexp_{ij}$  shows that the higher degree of matching of the other three classes, the lower the value. The range of the above four degree of interest is  $[0, 1]$ .

## 4. Experiment and Result Analysis

### 4.1. Sources and Processing of Data

In this section, we use the first order predicate logic to show the effectiveness of the association rules in the process of removing redundant rules, and the algorithm is implemented in Java. Experimental environment: Inter(R) Core(TM) i3 CPU M 330 @2.13 GHz, RAM 2GB, Windows 7 operating system. Operating tools: Weka (Waikato Environment for Knowledge Analysis Version 3.6.12). Data set: Mushroom from the UCI Machine Learning Repository (<http://archive.ics.uci.edu/ml/>).

Table 1 gives examples of these data sets and attribute features.

Table 1. Data set feature

| Relation  | Instances/(numbers) | Attributes/(numbers) |
|-----------|---------------------|----------------------|
| Mushroom  | 8124                | 23                   |
| Nursery   | 12960               | 9                    |
| Bank-full | 45211               | 17                   |

Take the Mushroom Database as an example. The Mushroom database contains 8124 records and 23 property items, including edible, mushroom cap shape, mushroom cap surface, mushroom cap colour, and so on for a total of 23 kinds. The association rules mining tools is Weka 3.6 in this experiment. After handling the original data from the Mushroom Database as the format file of csv, it was taken into the Explorer function from Weka 3.6 for Apriori association rules mining.

Mining of the specific parameters is set in Table 2:

Table 2. Weka 3.6 Association rule mining parameter setting

| Name                 | Value   |
|----------------------|---------|
| Associator           | Apriori |
| Car                  | False   |
| Class Index          | -1      |
| Delta                | 0.05    |
| LowerBoundMinSupport | 0.1     |
| Metric Type          | 0.9     |
| numRule              | 10000   |
| outPutItemSets       | False   |
| removeAllMissingCols | False   |
| Significance Level   | -1.0    |
| upperBoundMinSupport | 1.0     |
| Verbose              | False   |

After the mining task, we use Weka 3.6 to derive based on the confidence order of the 10000 rules. We have made a simple pre-treatment of the 10000 rules by removing irrelevant data and allowing 1000 rules to persist. The following experiments will be based on these rules. These rules are partly shown in Table 2 (the first 3 and the last 3 rules).

Table 3 instruction: Number of rules sorted by the value of the confidence value, and the range of confidence is [0, 1]. The front of the '=' is the class of property, and it is the value of property behind the '='.

Table 3. Weka 3.6 Example of derived association rule

| Number | Front-rules   | Behind-rules  | Confidence |
|--------|---|---|------------|
| 1      | veil-colour=white 4   | veil-type=partial   | 1          |
| 2      | gill-attachment=free  | veil-type=partial   | 1          |
| ⋮      | ⋮   | ⋮   | ⋮          |
| 9999   | gill-attachment=free<br>gill-size=broad<br>stalk-surface-below-ring=smooth 2<br>veil-colour=white 4 | stalk-surface-above-ring=smooth 1<br>gill-attachment=free | 0.97       |
| 10000  | gill-attachment=free<br>gill-spacing=close<br>gill-size=broad<br>stalk-surface-below-ring=smooth 2  | stalk-surface-above-ring=smooth 1<br>veil-colour=white 4  | 0.97       |

#### 4.2. Results of Experiment

By using different support and confidence of the data set, we obtain the association rules as the initial data and delete redundant association rules. Then, we calculate the degree of interest after deleting the data. From the results of  $unexpConseq_{ij}$ 、 $unexpCond_{ij}$ 、 $bsUnexp_{ij}$  and  $Confm_{ij}$ , we finally pick out the association rules that meet the requirements



as the experimental results.

Setting numRule 10000, the LowerBoundMinsupport is 10% and the MinMetric is 90% from the data set of mushroom. The LowerBoundMinsupport is 15% and the MinMetric is 85% from the data set of Nursery. The LowerBoundMinsupport is 15% and the MinMetric is 90% from the data set of Bank-full.

The results are shown in Table 4:

|            | Mushroom | Nursery | Bank-full |
|------------|----------|---------|-----------|
| Numbers    | 10000    | 10000   | 10000     |
| Redundancy | 8673     | 6532    | 8482      |
| Interest   | 18       | 25      | 33        |

The subjective interest optimization algorithm involves more subjective content, and it is difficult to evaluate the results from the objective aspect. However, we can achieve the user's ideal experimental results by adding a priori knowledge of the user.

## 5. Conclusions

Association rule mining is faced with the problem of high quantity and low quality. There is important research significance to optimize the rules that have been found. We further improve subjective interest from the perspective of the optimization algorithm, and we combine the first order predicate formula to remove the redundant association rules algorithm and the subjective interest degree algorithm. Through experimental verification, the improved subjective interest rate analysis algorithm has been effectively enhanced.

## Acknowledgement

This article was supported by the National Natural Science Foundation of China under Grant No. 71461017.

## References

1. B. Liu, W. Hsu, S. Chen, and Y. M. Ma, "Analysing the Subjective Interestingness of Association Rules," *Intelligent Systems and Heir Applications*, Vol. 15, No. 5, pp. 47-55, 2000
2. G. Piatetsky-Shapiro and C. J. Matheus, "The Interestingness of Deviations," *U. M. Fayyad, R. Uthurusamy, Knowledge Discovery in Databases. Seattle: AAAI Press*, pp. 25-36, 1994
3. Y. Zhen and Q. X. Zhu, "Improvement and Application of FP-TREE Algorithm based on User Interest," *Computer Engineering and Applications*, Vol. 8, No. 11, pp. 143-147, 2012
4. X. Z. Niu, J. Yang, and M. T. Zhou, "The Optimization Algorithm of Association Rules based on the Subjective Interest," *Journal of Sichuan University: Engineering Science Edition*, Vol. 45, No. 4, pp. 131-139, 2013
5. S. Y. Wei, G. L. Ji, and W. G. Qu, "Redundant Deletion and Clustering of Association Rules," *Journal of Chinese Mini-Micro Computer Systems*, No. 27, pp. 111-113, 2006
6. M. K. Deng and X. C. Shao, "Discrete Mathematics," *Tsinghua University Press*, 2014
7. C. Y. Yuan, "Petri Net Application," *Science Press*, 2013
8. H. Y. Wang, Q. Huang, C. T. Li, and B. Z. Zhu, "Graph Theory Algorithm and its MATLAB Implementation," *Beihang University Press*, 2010

**Xiaodong Qian** graduated with a doctorate from the School of Management Science and Engineering in Tianjin University. He entered the post-doctoral mobile station of Control Science and Engineering in Tianjin University. He is currently a professor at Lanzhou Jiaotong University and a member of CCF. His current research interests include data mining, data analysis, and business intelligence.

**Rui Guo** is a Master's student from the School of Electronics and Information Engineering in Lanzhou Jiaotong University. His research interests include data mining and intelligence algorithms.