Research on Optimization of Association Rules Mining Algorithm

Ningning Li and Zhaoxia Lu
Shandong Sport University, Jinan Shandong Province, China, 250102

Abstract. The association rules mining algorithm is always one of the specialists most concerned topics in the research of database knowledge. It is widely used in many fields in society. This article is based on association rule mining algorithm to put forward the relevant strategies for the majorization to make the mining algorithm get better application. With the development of the society, social information becomes richer and richer, people also become more and more interested in the unknown world. Sustainably improve the automation and intelligentization of the tools for processing information can continually improve the abilities in many ways and get more useful information from the huge database. So the appearance of the data mining makes it possible. It can fetch some hidden, unknown and useful information and knowledge from the huge practical application data. And the association rule is a main mode in research of data mining. It can fetch target data by finding the dependency between fields of a given condition from different data fields. As a result, the optimization of association rules mining algorithm is of quite importance in fetching target data.

1. The Related Concept and Reference Model of the Association Rules

Data mining refers to the mining of some “tacit” knowledge hidden in massive data, in order to help decision makers make decisions. Generally speaking, there are mainly such types of mining models, namely, concept description: characterization and comparison, association rule mining, classification and prediction, cluster analysis, outlier analysis and evolution strategy. Association rule mining is one of the focuses for our study, which is the basic problem and the most important issue in data mining research. Association rule mining is to dig out the relationship between projects from a large transaction database. The process includes finding frequent itemsets and generating association rules. Currently, we can find that we mainly focus on finding frequent itemsets. Agrawal has conducted in-depth research on this aspect and proposed the Aprior algorithm. This classic algorithm is of great benefit in later generations. Many of our current algorithms are derived and developed based on the Aprior algorithm. Meanwhile, the later generations have long-term exploration in theory and practice and make the Aprior algorithm improved. It is a fact that there are still some defects in the Aprior algorithm itself, which has not been completely revised by the later generations. Therefore, a number of scholars and experts have carried out research and improvement on many aspects, which further improve the effectiveness of association rules mining.

The association rule pattern is the most active one in data mining system. The initial generation of this model is proposed for market basket analysis, which aims to discover the relationship between different items in the transaction database and help to understand the customers’ purchase behavior model. It plays an important role in the production arrangement, merchandise shelf design and marketing.

Beer and diapers, a typical case, can well explain the function and significance of association rules. In the United States, the father who has a baby will go to the supermarket to buy diapers after work. By
mining information about customers' purchase behavior, supermarkets found that 30%-40% of these fathers would buy some beer after buying diapers. On the basis of this finding, supermarkets have adjusted the placement of commodity shelves, putting baby diapers and beer together, which results in a significant increase in beer sales. This case directly shows the connotation of association rules: a certain percentage of the customers who will buy product C and D after purchasing product A and B.

After Agrawal put forward the association rules, an in-depth analysis and research on mining problem have been made, which gradually improve the practical value of the association rules. While playing a role in guiding business, the functional utility of association rules mining has gradually penetrated into education, medicine, scientific research and other aspects, giving full play to the good mining effect.

In historical transaction database \( D= \{T_1, T_2, \cdots, T_m\} \), \( T_i \) means a transaction record and all the transaction records will combine into sets \( T_i \). Each term of the set \( T_i \) will be defined as a commodity, it will be represented with \( I_i \). When \( T_i=\{I_1, I_2, I_3\} \), it means the client bought three commodities which marked \( I_1, I_2, I_3 \) in a transaction at a time. And then define all the terms as a step \( T=\{I_1, I_2, \cdots, I_n\} \). In such situations, each record \( T_i \) is a subset of the set \( I \), so it can be expressed by \( T_i \subseteq I \). In addition, the itemset can be defined as the combination of \( k \) terms, and it can be indicated by \( k \)-terms \( \subseteq I \). In this case the \( k \)-terms not necessarily to be the Ti.

Preset two itemsets \( X \) and \( Y \), assuming \( X=\{x_1, x_2, \cdots, x_n\} \), \( Y=\{y_1, y_2, \cdots, y_n\} \), in this situation, the \( x_i \) or \( y_i \) is just one of the terms. If \( i\neq j \), \( x_i \neq x_j \), \( y_i \neq y_j \), and \( X \cap Y = \emptyset \). The definition of support to \( X \) is \( \text{supp}(X) = \{\text{the number of transaction records which includes } X \text{ in database } D\} \). It means when presets \( i=0 \), scanning backward from the first record in database \( D \), all the records which contains \( X \) can be expressed by \( i++ \) and finally the supp(\( X \)) is the result value of \( i \). The association rule is an implication \( XY \), it represents the incidence relation between \( X \) and \( Y \). If show it by confidence the degree of association is a conditional probability which means \( \text{conf}(XY)=P(Y|X)=p(XY)/P(X)=\text{ supp}(XY)/\text{ Supp}(X) \). Setting a minimum support threshold which expresses with \( m \) in supp and a minimum confidence represents by \( \text{min} \_\text{conf} \) and then when the support of an itemset is not less than a minimum support threshold, it becomes a frequent itemset. When the confidence of the association rule is not less than a minimum confidence, it can be the strong association rule.

From the above information, the mining of the association rule is a process of fetching all the frequent itemsets, then deriving a strong association rule from them and finally outputing it.

2. The Optimization Strategies of Association Rule Mining Algorithm

2.1. The Data Segmentation Optimization

The definition of the data segmentation is deviding the database \( D \) into \( n \) non-adjacent data segments \( D_1, D_2, \cdots, D_n \), the global minimum number of database supports is \( \text{min} \_\text{count} \), the local minimum support number \( \text{min} \_\text{count}i(i=1,2,\cdots,n) \) for data slice \( Di \) can be expatiated on \( \text{min} \_\text{count}i=\text{min} \_\text{count}*llDill/llDill \).

When the propositional database \( D \) be devided into \( n \) non-adjacent data segments \( D_1, D_2, \cdots, D_n \), and there is a project sequence \( IS \) of items not very frequent in all data segments \( Di= (i=1, 2, \cdots, n) \), as a result, it is not frequent in global databases either.

Verification: assuming there is a project sequence \( IS \) not frequent in all data segments defined as \( Di \) (\( i=1, 2, \cdots, n \)), the support number in \( Di \) is \( s \_\text{counti} \) (\( IS \)). And because \( IS \) is not frequent in all data segments which defined as \( Di \) (\( i=1, 2, \cdots, n \)), as a result, there is a relation that \( s \_\text{counti} (IS) < \text{min} \_\text{counti} \). Besides, since the support number of \( IS \) in \( D \) which defined as \( s \_\text{counti} (IS) \) Is the sum of its support numbers across the entire database, it is calculated as follows:

\[ s \_\text{counti} (IS) = \sum s \_\text{counti} (IS) < \sum \text{min} \_\text{counti} = \sum (\text{min} \_\text{counti}*llDi/llDill)= \text{min} \_\text{counti} * (\sum llDi/llDill)= \text{ min} \_\text{counti} \]

Hence: \( s \_\text{counti} (IS) < \text{min} \_\text{counti} \), it shows that \( IS \) is not frequent in total databases \( D \).

So we can change the process of ISS-DM (\( Di, \text{min} \_\text{counti} \)) into a callable process. With the transformation, the Suboperational operation of the itemized sequence set can be used to finish the evolution of local frequent item sequences. Specific description is as algorithm 1: Procedure ISS-DM (\( Di, \text{min} \_\text{counti} \))
1. ISS(\text{?})
2. FOR all IS < Di DO BEGIN
3. join(IS, ISS);
4. make_fre(IS, ISS, ISS*, min_count_i);
5. END;

This procedure tests the support numbers of the target IS by testing every tuple of database to fetch the corresponding sequence of items IS and finally with the subprocesses which defined as join(IS, ISS) to achieve it. Moreover, lording the support number into ISS specifically and then with the subprocesses defined as make_fre (IS, ISS, ISS*, min_count_i) to get corresponding local frequent item sequence set. When generating a locally frequent item sequence set, the local minimum number min_count_i can be used to finish it and in this case the i can be 1, 2, \ldots, n.

In this process the local frequent item sequence sets can just be derived from ISS*. If you want to get the global frequent item sequence sets, you should scan the database twice and recount the support numbers of these local frequent item sequence sets with the global minimum supporting number. In order to achieve it, we can design the procedure Generate_gDISS (D, min_count_i, ISS*) which can be called algorithm 2 and its specific description is as follows:

Procedure Generate_gDISS (D, min_count_i, ISS*)
1. FOR all IS*(ISS*) DO s_count(IS*) = 0;
2. FOR all IS(D) DO BEGIN
3. FOR all IS*(ISS*) DO
4. IF IS* (sub{IS}) THEN s_count(IS*)++;
5. END
6. FOR all IS*(ISS*) DO
IF s_count (IS*) < min_count_i THEN
ISS* = ISS*—{IS*};

2.2. The Optimization Processing of Algorithm

The smaller the database is, the less time will be spent in scanning, and the more efficient it will be. In specific procedure, the optimization of algorithm can be used to realize the goal of shrinking the database. It means that the information which is useless to generate the maximum set of items can be deleted by trimming database. So it can improve the efficiency of calculating support degree of candidate item set. The following discuss the issue of trimming database from two aspects which one is to prune the database transaction size and the other one is to reduce the number of transaction in database itself.

In terms of trimming database transaction and the definition that any subset of the maximum project item sequence set is also the maximum project item sequence set, we can find if there is a transaction contains (K+1)-dimensional maximum itemsets it will be no less than the number (K+1) of the K-dimensional maximum itemsets in last program loop. And all the sets included in these project itemsets will appear at least K times in K-dimensional candidate project set. So we can find the result that if the (K+1)-dimensional maximum itemsets are contained in this transaction or if some project sets are contained in (K+1)-dimensional maximum itemsets with the judgement that if some K-dimensional subsets in a transaction are included in candidate itemsets or whether it appears K times in K-dimensional itemsets or not. Otherwise, the useless information to later generate the largest project itemsets in transactions can be deleted. In this way, we can efficiently shrink the searching database and improve the efficiency of searching.

However, all above are just necessary conditions. Once the K-dimensional project itemset is the subset of the transaction called t and it is also contained in candidate project itemset C_k, the support number of the itemset will plus one. If a [I] is used to express the frequency of the first item in transaction t, it will plus one when this itemset includes the first item. And compare the size of a [I] after processing all candidate item sets. If a [I] < K, the first item in the transaction will be discarded.

When reducing the number of database transactions, a transaction can be discarded if any item of it not appear in any maximum project items in any cycle. And if the item number of the transaction was less than K, it can also be pruned. In this way we can efficiently narrow down the data searching. The improved description of algorithm is as follows:
(1) L_1 = \{frequent 1—itemsets\};
(2) discard the items which are not contained in 1—itemsets in database;
(3) For \{K=2; L_k \neq \emptyset; K++\} do begin
(4) C_k = \text{apriori-gen}(L_{k-1});
(5) for all transaction \(t \in D\) do begin
(6) C_1 = \text{subset}(C_k, t);
(7) for all candidates \(C \in C_t\), do
(8) \(C . \text{count}++\);
(9) \(t . a[I]++\);
(10) End
(11) for all itemset \(\in t\) do begin
(12) If \(t.a[I] < k\) then // \(a[I]\) means the frequency of the first item in transaction in \(L_k\);
(13) Prune the item;
(14) End;
(15) If (the \(t\) not include any frequent item in \(L_k\) or the item number of \(t\) is less than or equal to \(k\), then
(16) Delete the set \(t\) from the database
(17) End
(18) \(L_k = \{C \in C_k \mid C . \text{count} \geq \text{minsupport}\}\)
(19) End
(20) Answer = \(U_k L_k\)

By the way of pruning database makes the process of calculating the support greatly simplified and the efficiency of the algorithm efficiently improved.

3. Conclusion
There are many ways of association rules mining algorithm, but each one has its specific condition. So, if we want to optimize the algorithm, we must fully understand its characteristics and applicable conditions to find the point and direction of optimization. In general, recently the research of association rules mining algorithm is still at the primary stage. There are also many unsolved problems, so the association rules mining algorithm deserves further study. We should improve the algorithm while considering the usage requirements and data mining quality, and strive to visualize the association rules, making the results visual, intuitive and easy to use. Association rules data mining technology is in the process of continuous development. With the emergence of new situations, the application of association rules technology will be more extensive.

4. Reference
[1] Yuan Xiaoling, Zhao Qian, Gui Zhenmei. The optimization of association rule mining algorithm [J]. Journal of the Hebei Academy of Sciences, 2015(2)
[2] Li Xiaobing, Wu Jinlin, Xue Yongsheng, Weng Wei. Research on improvement and optimization of association rule mining algorithm [J] Journal of Xiamen University (Natural Science), 2005(4)
[3] Zhang Qing. The optimization of association rule mining algorithm [J]. Henan Science, 2015(1)
[4] Zuo Xiangke, Xing Yongkang, Wang Rong. Research on association rule mining algorithm and optimization strategy [J]. Microprocessors, 2009(5)