An Association Rule Algorithm for Online e-Commerce Recommendation Service

Jiabei Dai and Bin Zeng

***Abstract*—A novel technique of association rules to provide efficient recommendation services for E-Commerce environment is proposed in this paper, which is used to help online shop managers to increase profit and give associate product recommendation to online customers. In order to reach these two goals this technique should be based on profit and give a better recommendation to buyers. Here the profit-support association rule algorithm is presented, which uses a unique profit to generate a minimum support for every item and multiple minimum supports to mine association rules. Through several experiments, we have shown that these optimization techniques can yield significant performance improvement.**

***Index Terms*—Association rules, data mining, information extraction, recommendation system.**

I. Introduction

Recently, many of commerce systems are willing to apply association rule mining to their businesses. They have been trying to increase their own profits and provide customer services. Particularly, decision makers want to increase the precise investment costs and also returns of the investment (ROI) by applying such mining systems. However, it is difficult for the traditional association rule algorithms to meet such requirements. Thus, we note that such traditional association rule algorithms are simply based on sales volume of items (i.e., whether a certain item is included in a particular transactions). It means that it is impossible to discover the items of whose sales volume are rare even though they are very profitable. On the other hand, in terms of recommendation services, customers will obtain the items which are frequently occurred in many transactions. Then, the chance of increasing the profit might be getting lower with these services.

Though there are kinds of information systems relating to how to manage market transactions, market managers still expect to apply information processing techniques for better business performance (e.g., reducing costs, increasing profits and providing users with better services). In order to satisfy the requirements of market managers, some commerce recommendation systems are introduced to apply association rule mining to efficiently conduct such management tasks.

In the literature of data mining, a plenty of different algorithms for association rule mining have been proposed [1]-[3]. However, they can not support decision makers to discover useful business strategies in practice. Association rule mining on the problem of shop basket analysis is especially based on the history of customer purchase records in commerce environments [4]-[6].

However, we have realized there is a problem about applying the existing association rule algorithms to the market recommendation service. Most of the rules mining schemes such as Apriori algorithm consider only the sales volume of the items while ignoring the profit of items. Thereby, we proposed a novel algorithm, profit-support association rule algorithm, to adopt the relationship between the sales volume and the profit of each item during the data mining process.

The outline of this paper is organized as follows. In Section II, we introduce back-ground and the previous work on association rules. Section III describes the problem statement of this work. In Section IV, we present profit-support association rule mining method to deal with this problem. Section V shows a case study that we have tested the proposed method for evaluation. The conclusion is addressed in the Section VI.

II. Related Research

Given a set of transactions *T* (database), the problem of mining association rules is to discover all association rules that have support and confidence greater than the user-specified minimum support (*minsupp*) and minimum confidence (*minconf*). Most of the association mining algorithms work in two steps:

1) generating large itemsets that satisfy *minsupp*,
2) generating association rules that satisfy *minconf* using the large itemsets.

Association rule mining has been studied extensively in the previous work [7]-[9]. The models used in these studies are quite same in terms of finding all possible association rules that can meet user-specified constraints (i.e., *minsupp* and *minconf*).

Association rules are playing an important role in data mining-based applications. One of the well-known classic applications is market basket analysis [10]. It can analyze a set of transactions (also called itemsets) purchased by customers, and discover meaningful patterns (e.g., association among items). Roughly speaking, an association rule can be regarded as a relationship in the form of $A \rightarrow B$, where A and B are two distinct items. Two measures, namely the support and the confidence, are used to refine a rule. While the support is the percentage of transactions contained both A and B in the whole data set, the confidence is the ratio of the number of transactions that contain A and B over the

Manuscript received July 4, 2016; revised September 7, 2016.
Jiabei Dai is with the University of British Columbia, Canada (e-mail: zbtrueice@163.com).
Bin Zeng is with the Department of Management at the University of Naval Engineering, China.

doi: 10.18178/joebm.2016.4.10.454

total number of transactions that contain A. For example, suppose that many people who purchase 'DVD Player' and DVD Disk' also tend to purchase 'Beer'. The corresponding association rule can be "DVD Player ∩ DVD Disk. → Beer".

III. Problem Analysis

The key element that makes association rule mining practical is m. is *minsupp*. It is used to prune the search space and to reduce the number of rules generated. However, in previous algorithms [11]-[13], they only use single *minsupp*. It means that the studies implicitly assume that all items in the database are of the same nature (to be explained below):

1) Every item has the same profit.
2) Every item has similar frequencies in the database.

This is not suitable for the case in real world. Firstly, in terms of profit, even though the sale of some items have occurred only a few times (less than the predefined *minsupp*), they can be more important (e.g., much more expensive) than the others which have occurred more frequently. For example, CD Player is much more expensive than CDs. If using previous association rule mining algorithms (e.g., Apriori algorithm [2]), we will encounter two problems: first, we always mine out some rules making few profits. Second, in the first iteration of the Apriori algorithm to generate 1-itemset some items are deleted, which can make higher profits but have lower support. For example, if *minsupp* = 2%, then two rules can be potentially discovered:

1) Rule A: DVD Disk (support=3%) → DVD Player (support=0.5%)
2) Rule B: DVD Disk (support=3%) → Beer (support=4%)

Clearly, Rule A can make more profit. But we only can discover Rule B if we use the previous algorithms.We have to remove these items in the first iteration to generate 1-itemset. Since the goal of our data mining task is to increase the profit, we should consider profit when we conduct association rule mining not only concerning the amount of items. Secondly, in many applications, some items appear very frequently in the data, while others rarely appear. If the frequencies of items vary highly we will encounter two problems:firstly, if *minsupp* is set too high, we will not find those rules that involve infrequent items or rare items in the data. Secondly, in order to find rules that involve both frequent and rare items, we have to set *minsupp* very low. However, this may cause combination explosion, producing too many rules, because those frequent items will be associated with one another in all possible ways and many of them are meaningless. This dilemma is called the rare item problem [14]. When confronted with this problem in applications, researchers either split the data into a few blocks according to the frequencies of the items and then mine association rules in each block with a different *minsupp*, or group a number of related rare items together into an abstract item so that this abstract item is more frequent. The first approach is not satisfactory because rules that involve items across different blocks are difficult to be found. Similarly, the second approach is unable to find rules involving individual rare items and the more frequent items. Clearly, both approaches are adhoc and approximate. One efficient approach to solve this problem is to design an algorithm which tries to mine association rules with multiple minimum supports. In this paper we will use this approach mining association rules with N-itemsets.

IV. Algorithm Description

This algorithm derived from the Apriori algorithm. The difference between this papers and previous approaches is that we use the profit as a criterion to set minimum support for each item [15l, but previous algorithms use the percentage, the amount of items, to set minimum support. So in our algorithm for finishing the whole process we first set minimal support for each item using unique profit criterion. Then we mine large itemsets with multiple minimal supports.

A. Generation of 1-Itemset

In previous data mining algorithms they use the percentage of items in the transaction database as a criterion to evaluate the importance of each item. As mentioned before there are two implicitly assumes that all items in the data are of the same nature. Every item has the same profit and every item has similar frequencies in the database. But it is not the fact in real life shopping transactions. Actually for super market and online chain store managers their final goals are to enhance the profit from the business.

Therefore from the beginning of association rule mining we put our attention on profit attribute. We use minimum total profit to generate 1-itemset. In this step we only compare the total profit of each item with the minimum total profit. Because the items with low total profits take less weight of advantage even they have large amounts. We delete all the 1-items which have total profits less than minimum total profit. For example, in Table I we delete item4.

TABLE I: Example of Setting Minimum Support for Item

	Item 1	Item 2	Item 3	Item 4
Amount of items	4	40	150	150
Unit profit	100$	2.5$	0.5$	0.5$
Total profit	400$	100$	75$	75$
Minimum total	100$			

B. Minimum Support for Each Item Using Profit

In this step we use Unique Profit to reevaluate and set Profit-support Amount, the minimum support, for each item and generate N-itemset. After generating 1-itemset the previous level-wise algorithms use one minimum support to do mining. They assume all items have the similar amounts. Obviously it is not the fact. To evaluate the importance of each item, profit is the most clear and effective character. So profit should be the criterion. But the profits of items float in a large range with the amount of items. For example, in the real supper market basket analysis beer is one frequent item and has large amount but low profit. The DVD Player is one infrequent item comparing to beer but has higher profit. From this point of view we use profit as a unique criterion to reevaluate the amount of items. In other words, here the profit margin is looked on as a power of amount of items. After reevaluating by profit, the amount of the item which has large amount but lower profit decreases corresponding to the profits. And the amount of the item which has little amount but higher profit increases. Another benefit getting from reevaluation of the amount is that we can still use the level-wise Apriori algorithm to do association rule mining which is based on the amount of items as threshold. This algorithm looks like the previous approaches. But, in fact, it is an approach using profit criterion. Before giving details we first introduce one definition:

Definition 1 (Appointed Profit) *lt is a unique profit value given by user and is used to calculate profit-support amount of each item.*

Table I shows an example of determining minimum support for items. In general, it is set as an integer and same as or times of the highest unit item profit. Because setting by this way most of the final profit-support amount can be a integer thereby preprocess the computation.

Definition 2 (Minimum Profit Support (MPS)) *Instead making the minimum support of all items to be the same, we want to consider the profit impact on the frequencies. The MPS is computed by*

$$MPS = \frac{Appo\,\mathrm{int}\,ed\ \ profit}{Unit\,profit} \tag{1}$$

For example, in Table 1, minimum profit support of Item2 is calculated as follow, suppose Appointed Profit is 100

$$MPS(Item2) = \frac{Appo\,\mathrm{int}\,ed\ \ profit}{Unit\,profit} = \frac{100}{2.5} = 40 \tag{2}$$

It means that in order to get 100$ profit by selling Item2, we have to sell 40 Item2. In the same way we calculate minimum profit support of Item l. The result is equal to 1. It means if we sell one Item1 we can get 100$ profit. Thus, we can get the same profit we should sell different amount of Item1 and Item2. From this simply example we can find that profit-support algorithm is more reasonable.

C. *Mine of Large Itemsets*

After setting MPS, for each item, we can discover large itemsets with MPS as describe in [5]. Like Apriori algorithm, the proposed algorithm is also dependent on level-wise searching. It generates all large itemsets by making multiple passes over the data. In the first pass, it counts the supports of individual items and determines whether they are large. In each subsequent pass, it starts with the seed set of itemsets found to be large in the previous pass. It uses this seed set to generate new possibly large itemsets, called candidate itemsets. The actual supports for these candidate itemsets are computed during the pass over the data. At the end of the pass, it determines which of the candidate itemsets are actually large. However, there is an important exception in the second pass, as we will consider later. A key operation in the proposed algorithm is the sorting of the items in the set of items in ascending order of their MPS. This ordering is used in all subsequent operations of the algorithm. The items in each itemset also follow this order. For example, in Table I, there are four items. *MPS*(Item1) = 10%, *MPS*(Item2) = 20%, *MPS*(Item3) = 4% and *MPS*(Item4) = 4%. Finally, they can be sorted as follows: Item3, Item4, Item1, and Item2.

V. Experimental Results

In this section we show some experiments to evaluate the performance using in e-commerce environment and the efficiency of the proposed association rule mining algorithm (PARMA). We show that the algorithm allows us to find rules with very low supports, even involved rare items, yet without generating a large number of meaningless rules with frequent items.

In our experiment, similar to [7l, we have used IBM synthetic data generator to generate the data set with the following parameters: 1,000 items, 10,000 transactions, 10 items per transaction on average, and 4 items per frequent itemsets on average. Also, we have decided the profit of each single item as follows; 80% of items have a medium profit ranging from $1 to $5, 10% of items have a high profit ranging from $5 to $10, 10% of items have a low profit ranging from $0.1 to $1. This is a simplified version of the normal distribution. The exact profit of each item is determined by random selection from the respective profit range. We have considered only single quantity sales for each item.

In our experiments we have evaluated the performance of the proposed algorithm by comparing to amount-based association rules mining with multiple minimum support, MMSapriori [2l. Both algorithms are commonly based on multiple minimum supports.While MMSapriori is simply making quantities of items as the optimization goal, our proposed approach considers the profit of each item.In fact, the proposed algorithm is partially based on MMSapriori. It means the way of calculating N-itemsets (N > 2) with multiple minimum supports is the same. In other words, the only difference is how to generate 1-itemset. Thus, as shown in Table II and Table III, these two algorithms have been compared.

TABLE II: Number of Large Itemsets Found. LS Stands for the Lowest Support

	MMSapriori			PARMA		
LS	a=2	a=10	a=20	a=2	a=10	a=20
0.1%	5140	25030	29570	5070	24000	28470
0.2%	4920	12850	13100	4750	11250	12020
0.3%	2400	5040	5910	2300	4980	5900

TABLE III: Number of Candidate Itemsets

	MMSapriori			PARMA		
LS	a=2	a=10	a=20	a=2	a=10	a=20
0.1%	326520	356840	402560	345450	369510	422100
0.2%	269540	275100	290110	285640	285000	310050
0.3%	225460	235420	241050	235010	250550	259840

We found that the results of our proposed PARMA algorithm are always lower than that of MMSapriori as shown in Table II, since PARMA can remove most of 1-items which take less profit at the first time to pass over the database. Moreover, while the numbers of large itemsets by MMSapriori are mostly larger than by PARMA, PARMA generates more candidate itemsets as shown in Table III. Thus, it proves that PARMA is more efficient than MMSapriori. Firstly, PARMA uses minimum total profit value to generate 1-itemset at the beginning of the process so that it can delete most of the useless items. Secondly, we use profit support to generate N-itemsets. This step deletes the itemsets which take less weight of profits efficiently.

VI. Conclusion

Profit-support association rule for E-commerce improves the efficiency of user operation and helps the online shop managers to get more profits not only to increase the quantity of sale. In order to make such association rules truly practical and efficient, in this study, we first introduce unique profit criterion to generate unique minisupp for each item and combine this technique with the traditional algorithm of mining association rules with multiple minimum supports. Finally, we apply this profit-support association rule mining algorithm in E-commerce environment.

Our algorithm, Profit-support Association Rule algorithm, still needs to solve some problems. For example, the minimum total profit is difficult to be set value, the time-consuming is very long and how we improve the efficiency. We should modify this approach to solve these problems in the future.

References

[1] S. C. Young, C. M. Song, and H. R. Keun, "Mining association Rules using RFM scoring method for personalized u-commerce recommendation system in emerging data," in *Proc. the 2nd Communications in Computer and Information Science*, pp. 190-198, 2012.

[2] J. Dongre, G. L. Prajapati, and S. V. Tokekar, "The role of Apriori algorithm for finding the association rules in data mining," in *Proc. the 2014 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT)*, Ghaziabad, India: IEEE, pp. 657-660, 2011.

[3] Y. L. Chen and C. H. Weng, "Mining fuzzy association rules from questionnaire data," *Knowledge-Based Systems*, vol. 22, pp. 46-56, Jan. 2009.

[4] J. B. Woon, Y. S. Cho, and K. H. Ryu, "Personalized e-commerce recommendation system using RFM method and association rules," *KSCI*, vol. 15, pp. 227-235, Dec. 2010.

[5] J. Ronga, H. O. Vua, and R. Lawb, "A behavioral analysis of web sharers and browsers in Hong Kong using targeted association rule mining," *Tourism Management*, vol. 33, pp. 731-740, Apr. 2012.

[6] R. Belohlavek, J. Dvorak, and J. Outrata, "Fast factorization by similarity in formal concept analysis of data with fuzzy attributes," *Journal of Computer and System Sciences*, vol. 73, pp. 1012-1022, May. 2007.

[7] Y. S. Chen and C. H. Cheng, "A soft-computing based rough sets classifier for classifying Ipo returns in the financial markets," *Application of Soft Computation*, vol. 12, pp. 462-475, Apr. 2012.

[8] J. Jaishree, H. Ram, and J. S. Sodhi, "Improving efficiency of Apriori algorithm using transaction reduction," *International Journal of Scientific and Research Publications*, vol. 3, pp. 1-4, Jan. 2013.

[9] X. H. Li and S. Jin, "An improvement of the new Apriori algorithm," *Computer Science*, vol. 34, pp. 196-198, Apr. 2007.

[10] G. F. Kuang and Y. C. Li, "Using Fuzzy Association Rules to Design E-commerce Personalized Recommendation System," *Indonesian Journal of Electrical Engineering*, vol. 12, pp. 1519 -1527, Feb. 2014.

[11] A. A. Kardan and M. Ebrahimi, "A novel approach to hybrid recommendation systems based on association rules mining for content recommendation in asynchronous discussion groups," *Information Sciences*, vol. 219, pp. 93-110, Oct. 2013.

[12] S. X. Bai and S. L. Duan, "Association rule mining algorithm DB-growth based on relational database," *International Journal of Granular Computing, Rough Sets and Intelligent Systems*, vol. 4, pp. 105-121, Jan. 2015.

[13] H. Fan and K. Ramamohanarao, "Fast discovery and the generalization of strong jumping emerging patterns for building compact and accurate classifiers," *IEEE Transaction of Knowledge of Data Engineering*, vol. 18, pp. 721-737, Aug. 2006.

[14] M. Garcia-Borroto, J. Martinez-Trinidad, and J. A. Carrasco-Ochoa, "A survey of emerging patterns for supervised classification," *Artificial Intelligence Review*, vol. 42, pp. 705-721, Aug. 2014.

[15] E. R. Omiecinski, "Alternative interest measures for mining associations in databases," *IEEE Transaction of Knowledge Data Engineering*, vol. 15, pp. 57-69, Aug. 2003.

Jia Bei Dai was born in Wuhan, Hubei, China in 1997. She is perusing her economic degree at the school of arts in University of British Columbia, Vancouver, Canada. The focus of her research is in the areas of economic analysis, finance, customer relationship management, human resources management and organizational behavior.

Bin Zeng was born in Wuhan, Hubei, China in 1970. He is a teacher in the Department of Management at the University of Naval Engineering. He obtained his doctor's degree in computer architecture from the Huazhong University of science and technology, Wuhan, China. The focus of his research is in the areas of information management, computer network, industrial engineering and system engineering. He teaches and supervises undergraduate and master students.