Multiple Supports based Method for Efficiently Mining Negative Frequent Itemsets

Xiangjun Dong[*,1], Tiantian Xu[1], Yuanyuan Xu[2] and Xiqing Han[3]

[1]*School of Information, Qilu University of Technology, Jinan, Shandong, 250353, P.R. China*
[2]*College of Science, Northeast Forestry University, Harbin, Heilongjiang, 150040, P.R. China*
[3]*Academic affaires office, Shandong Institute of Commerce and Technology, Jinan, Shandong, 250103, P.R. China*

Abstract: Negative frequent itemsets (NFIS), which refer to frequent itemsets with non-occurring and occurring item(s) like ($a_1a_2\neg a_3a_4$), have become increasingly important in real applications, such as bioinformatics and healthcare management. Very few methods have been proposed to mine NFIS and most of them only satisfy the user-specified single minimum support, which implicitly assumes that all items in the database are of the same nature or of similar frequencies in the database. This is often not the case in real-life applications. Although several methods have been proposed to mine frequent itemsets with multiple minimum supports (MMS), these methods only mine positive frequent itemsets (PFIS) and do not handle NFIS. So in this paper, we propose an efficient method, called *e-msNFIS*, to efficiently identify NSP with MMS by only using the identified PFIS without re-scanning database. We also solve the problem of how to set up minimum support to an itemset with negative item(s). To the best of our knowledge, *e-msNFIS* is the first method to mine NFIS with MMS. Experimental results on real datasets show that the *e-msNFIS* is highly effective and efficient.

Keywords: Positive frequent itemsets, Negative frequent itemsets, Negative association rule, Multiple minimum supports.

1. INTRODUCTION

As an important supplement of positive association rules (PAR), negative association rules (NAR) considering both non-occurring and occurring item(s) can play a vital role in some applications [1-3]. For example, in healthcare, a patient who has missed a vital medical appointment may encounter serious health issues. A critical challenge in mining NAR is how to efficiently mine negative frequent itemsets (NFIS). There are very few methods to mine NFIS[1-7], such as e-NFIS[6] and Free-PNP[7]. These algorithms, however, only satisfy user-specified single minimum support, which implicitly assumes that all items in the database are of similar frequencies in the database. This is often not the case in real-life applications. So several methods have been proposed to mine frequent itemsets with multiple minimum supports (MMS) [8-13] that means each item in the database has its own minimum support, such as MSapriori[8], CFP-growth[9], CFP-growth++[10], MSB_apriori+ [11] and so on. But these methods only mine positive frequent itemsets (PFIS), and do not consider negative ones.

In this paper, based on MSapriori[8] and e-NFIS[6], we proposed an efficient algorithm, named E-msNFIS, to mine NFIS with MMS. The main idea of E-msNFIS is to discover PFIS with MMS by MSapriori first, and then to use the idea as e-NFIS to calculate the supports of negative candidate itemsets (NCIS) only based on the identified PFIS without additional database rescanning. We also propose a method to set up the minimum support to an itemset with negative item(s). To the best of our knowledge, this is the first algorithm to mine NFIS with MMS without rescanning databases.

The rest of this paper is organized as follows. Section 2 summarizes related works. The *E-msNFIS* algorithm is discussed in Section 3. Section 4 presents experimental results, followed by conclusions and future work in Section 5.

2. RELATED WORK

In this section, we discuss related work including some NFIS mining methods and some methods of mining frequent itemsets with MMS.

A few papers studied mining NFIS with single minimum support. The authors [1-3] mainly focus on mining NAR and PAR based on frequent and infrequent itemsets for the form of A(*e.g.* (a_1a_2))$\Rightarrow B$, $A\Rightarrow\neg B$ (*e.g.* $\neg$ (b_1b_2)), $\neg A\Rightarrow B$ and $\neg A\Rightarrow\neg B$. The two sides of rules are either all positive or all negative itemsets. Another form of NAR(*e.g.* $a_1\neg a_2\Rightarrow b_1\neg b_2$) which includes both positive and negative items in each side is used [4]. It introduced maximum support to exclude meaningless frequent itemsets and used different minimum supports respectively for positive and negative items to make mining more efficient. It can use Apriori or FP-growth method to mine NFIS. In a study [5], the authors imported a bit string for each node of frequent pattern tree to store the prefix. The research proposed a method to construct frequent pattern tree including positive and negative items and an

[*]Address correspondence to this author at the School of Information, Qilu University of Technology, Jinan, Shandong, 250353, P.R. China; Tel: +86-531-89631251; E-mail: d-xj@163.com.

A preliminary version of this article has been published in the Proceedings of International Conference on Information Technology and Management Innovation, 2013. (ICITMI2013) 386-390.

algorithm named MFPN based on the frequent pattern tree to mine frequent patterns. An algorithm named *e-NFIS* to mine NFIS based on PFIS and an efficient method to generate negative candidate itemsets (NCIS) and calculate the support of NCIS without additional database rescanning have also been proposed [6]. A study proposed a definition of NAR and an algorithm called Free-PNP, based on this definition to mine NFIS and NAR in databases [7]. An improved method, named GNAR, based on a taxonomy tree to diminish the huge computing cost of mining NAR has been introduced [14]. Moreover, the concepts of over-frequent itemsets to solve the problem of exploding number of NFIS have been discussed [15]. The procedure of mining NFIS incorporates parameters on min support, max support and the maximum number of negative item in the itemset. A study calculated the support and confidence of NAR based on the information of PAR [16]. It has also proposed an algorithm to mine both NAR and PAR from frequent itemsets. A definition of valid NAR and an Apriori-based algorithm named PNAR to mine all valid PAR and NAR have been highlighted. The authors used constraints to reduce search space [18]. It has also proposed a method to mine PAR from frequent itemsets and mine NAR from infrequent itemsets.

Some studies focused on mining frequent itemsets with MMS. A research presented a framework of MMS and an efficient method named MSapriori to find frequent patterns [8]. Another study introduced FP-tree-like structure called MIS-tree to store the information of frequent patterns and an efficient MIS-tree-based algorithm called CFP-growth to mine frequent itemsets with MMS without rescanning database [9]. Furthermore, an improved CFP-growth algorithm called CFP-growth++ by introducing four pruning techniques to reduce search space has been defined [10]. MSB_apriori+ algorithm to mine frequent patterns with MMS has been defined [11]. Research proposed a multiple level minimum support (MLMS) model and an algorithm named Apriori_MLMS to mine simultaneously both frequent and infrequent itemsets based on MLMS model [12]. An efficient algorithm to mine direct and indirect association patterns with MMS has been presented [13]. It can mine association rules based on frequent items and rare items. It can mine indirect association patterns based on infrequent itemsets. An efficient algorithm based on the Apriori approach to mine large itemsets with MMS under the maximum constraints has been proposed [19]. A study focused presented an efficient algorithm named MSDMFIA based on FP-Tree to mine maximum frequent itemsets with MMS without rescanning database [20]. It mines maximum frequent itemsets firstly and then calculates the minimum supports of maximum candidate frequent itemsets to mine association rules.

3. E-MSNFIS

3.1. Framework

Below is how E-msNFIS works. First, it mines all PFIS using MSapriori method [8] with multiple minimum supports. Second, NCIS is generated based on PFIS. Third, supports of these NCIS are calculated by converting them to calculating supports of corresponding PSP.

3.2 Problem Statement

Let $I= \{i_1, i_2,..., i_n\}$ be a set of n distinct literals called positive items, $NI=\{\neg i_1, \neg i_2,..., \neg i_n\}$ be a set of I's corresponding non-occurring items called negative items. i_1 is called the positive partner of $\neg i_1$, and $\neg i_1$ is called the negative partner of i_1. *TD* is a transaction database of variable-length transactions over I and each transaction is associated with a unique identifier *TID*. A set of distinct items from $I \cup NI$ is called an itemset. The number of items in an itemset A is the length of the itemset, denoted by $length(A)$. An itemset of length k is referred to as k-itemset. Each itemset has an associated statistical measure called support, denoted by s. For an itemset A, $s(A)=A.count /|TD|$, where $A.count$ is the number of transactions containing itemsets A in TD.

In multiple minimum supports model, to ease the task of specifying many minimum item support (MIS) values by the user, the same strategies as those proposed [8] to mine PFIS can also be applied to mine NFIS.

Let MIS($\neg i$) denotes the MIS value of item $\neg i$. We use actual frequencies of the items in the database as the basis for MIS assignments. The formula is as follows:

$$MIS(\neg i) = \begin{cases} \beta f(\neg i) & \beta f(\neg i) > LS \\ LS & Otherwise \end{cases} \qquad (1)$$

$f(\neg i)$ is the actual frequency of item $\neg i$ in the database. LS denotes the minimum MIS value of all items. β ($0 \leq \beta \leq 1$) is a parameter that controls how the MIS value for items should be related to their frequencies. If $\beta = 0$, we have only one minimum support, LS, which is the same as the traditional association rule mining.

The **minimum support** of a negative itemset $A= \{a_1, \neg a_2... \neg a_k\}$, denoted as $mins(A)$, is the lowest MIS value among the items in A, *i.e.*,

$mins(A)$=min [MIS(a_1), MIS($\neg a_2$),..., MIS($\neg a_k$)].

To itemsets A and a minimum support $mins(A)$, (1) if A only contains positive items and $s(A) \geq mins(A)$, then A is called a PFIS, and (2) if A contains negative items and $s(A) \geq mins(A)$, then A is called an NFIS.

Example 1: Consider four items a, b, c and d in a database. Their minimum item supports are:

MIS (a) =15% MIS ($\neg b$) =25% MIS ($\neg c$) =8% MIS (d) =10%

If we find the candidate 4-itemset $\{a, \neg b, \neg c, d\}$ has 11% of support, then it is an NFIS, because it satisfies its minimum support 8%(= MIS ($\neg c$)).

3.3 Generate NCIS based on PFIS

To generate all non-redundant NCIS based on PFIS, we used the same method as e-NFIS to generate NCIS [6]. The basic idea of generating a NCIS is to change any positive items i in a PFIS to its negative partner. In detail, for a k-itemset PFIS, its NCIS is generated by changing any m distinct item(s) to its (their) negative partner(s), m=1, 2,..., k-1. We do not generate the NCIS based on a PFIS when $m=k$ (k>1) because in real applications, this kind of NCIS is meaningless.

Example 2. To 3- itemset (*abc*), its NCIS includes:

$(\neg abc)$,$(a\neg bc)$,$(ab\neg c)$, when m=1;

$(a\neg b\neg c)$, $(\neg ab\neg c)$, $(\neg a\neg bc)$, when m=2.

3.4 Calculate the Support of NCIS

We use the same method as e-NFIS[6] to calculate the support of NSIC as follows.

Given $X=\{x_1, x_2,\ldots,x_p\}$, $x_k\in I$ (k=1,2,... p), $Y_q=\{y_1,y_2,\ldots,y_q\}$, *i.e.*, $\neg Y_q=\{\neg y_1,\neg y_2,\ldots,\neg y_q\}$, $y_k\in I$ (k=1,2,...,q), then

$$s(X\neg Y^q)=s(X\neg y_1,\neg y_2,\ldots,\neg y_q)$$
$$=s(\mathrm{X})-\sum_{i_1\subset Yq}s(Xy_{i_1})+\sum_{i_1,i_2\subset Yq}s(Xy_{i_1}y_{i_2})+\ldots$$
$$+(-1)^{q-1}\sum_{i_1,i_2,\ldots i_{q-1}\subset Y^q}s(Xy_{i_1}y_{i_2},\ldots y_{i_{q-1}})+(-1)^q s(XY^q). \quad (2)$$

The basic idea of Equation (2) is the inclusion-exclusion principle in set theory.

Example 3. $s(\neg a\neg bc)=s(c)+(-1)^1[s(ac)+s(bc)]+(-1)^2 s(abc)$

Especially, if an NCIS only contains one negative item, Equation (2) becomes

$$\mathrm{s}(\neg\mathrm{x})=|\mathrm{TD}|-\mathrm{s}(\mathrm{x}) \quad (3)$$

4. ALGORITHM E-MSNFIS

The e-msNFIS algorithm is proposed to mine NFIS using only identified PFIS as follows.

Algorithm: e-msNFIS;

Input: *TD*: database; MIS(*i*): MIS Value of each item;

Output: PFIS and NFIS;

```
(1) Use MSapriori algorithm to get PFIS and their supports;
(2)//Mining NFIS
(3) for each $X$ in PFIS do {
(4)     if length($X$)=1 then { /* $X$ only contains one item */
(5)         $s(\neg X)$=|TD|- $s(X)$;
(6)         if $s(\neg X) \geq$ MIS($\neg X$) then NFIS= NFIS$\cup(\neg X)$;
(7)     }
(8)     else {   /* $X$ contains more than one item */
(9)         NCIS = generates all candidates of $X$
(10)        For (each ncis in NCIS) do {
(11)            calculate $s(ncis)$ by Equation (2) ;
(12)            $mins(ncis)$ = min(MIS($ncis[a_1]$),..., MIS($ncis[\neg b_n]$)) ;
(13)            if $s(ncis) \geq mins(ncis)$ then NFIS= NFIS$\cup\{ncis\}$;
(14)        }
(15)    }
(16) }
(17) return PFIS and NFIS;
```

Line 1 mines all PFIS and stores their actual supports simultaneously from database using MSapriori algorithm with multiple minimum supports. Line 2 to line 16, we get NFIS based on PFIS. Line 4 to 7 generates the NFIS which only contains one item. Line 5 calculates the actual support of negative item $\neg X$. If $s(\neg X) \geq$ MIS($\neg X$), then $\neg X$ is inserted into NFIS(line 6). Line 8 to15 generates the NFIS containing more than one item. Line 9 generates NCIS based on PFIS by the approach presented in Section 3.3. Line 11 calculates the support of each *ncis* in *NCIS* by Equation (2). Line 12 calculates the minimum support of *ncis* that is the lowest MIS value among the items in *ncis*. If $s(ncis) \geq mins(ncis)$, then *ncis* is inserted into NFIS(line 13). Line 17 returns the results and ends the whole algorithm.

5. EXPERIMENTS

To evaluate the performance of e-msNFIS, 4 real datasets, *i.e.*, Mushroom, Chess, Connect, and Nursery, which can be obtained from http://archive.ics.uci.edu/ml/datasets.html underwent experiments. Because e-msNFIS is the first method to mine NFIS with MMS, there is no suitable algorithm to compare.

The algorithms of mining PFIS with MMS like MSapriori are not suitable to compare with e-msNFISbecause e-msNFIS firstly discovers PFIS with MMS by using MSapriori algorithm and then mines all NFIS based on identified PFIS. The runtime of e-msNFIS is the sum of PFIS's runtime and NFIS's and it is larger than PFIS's. PFIS and NFIS are not comparable in the number of NFIS. So both algorithms are not comparable both in runtime and the number of NFIS. The algorithms of mining NFIS with single minimum support like e-NFIS are also not suitable to compare with e-msNFIS because both algorithms mine PFIS in the first step and they mine different number of PFIS due to different methods to support setting. They mine all NFIS based on identified PFIS in the second step. Different number of NFIS is mined by both algorithms due to different number of PFIS. So both algorithms are not comparable both in runtime and the number of NFIS.

Our experiments were performed on a Pentium 4 Celeron 2.1G PC with 2G main memory, running on Microsoft Windows XP. All the programs were written in MyEclipse 8.5. Table **1** presents details of the datasets. For more details about the 4 real datasets, please refer to [11].

Table 1. Datasets for Run Time and Space Tests

Data Set	# of Trans	# of Items
Mushroom	8124	23
Chess	3196	9
Connect	67557	6
Nursery	12960	28

In the experiments, we set β = 0.6 and use different *LS* values to reflect the differences in 4 real datasets. The results of the experiments are as follows:

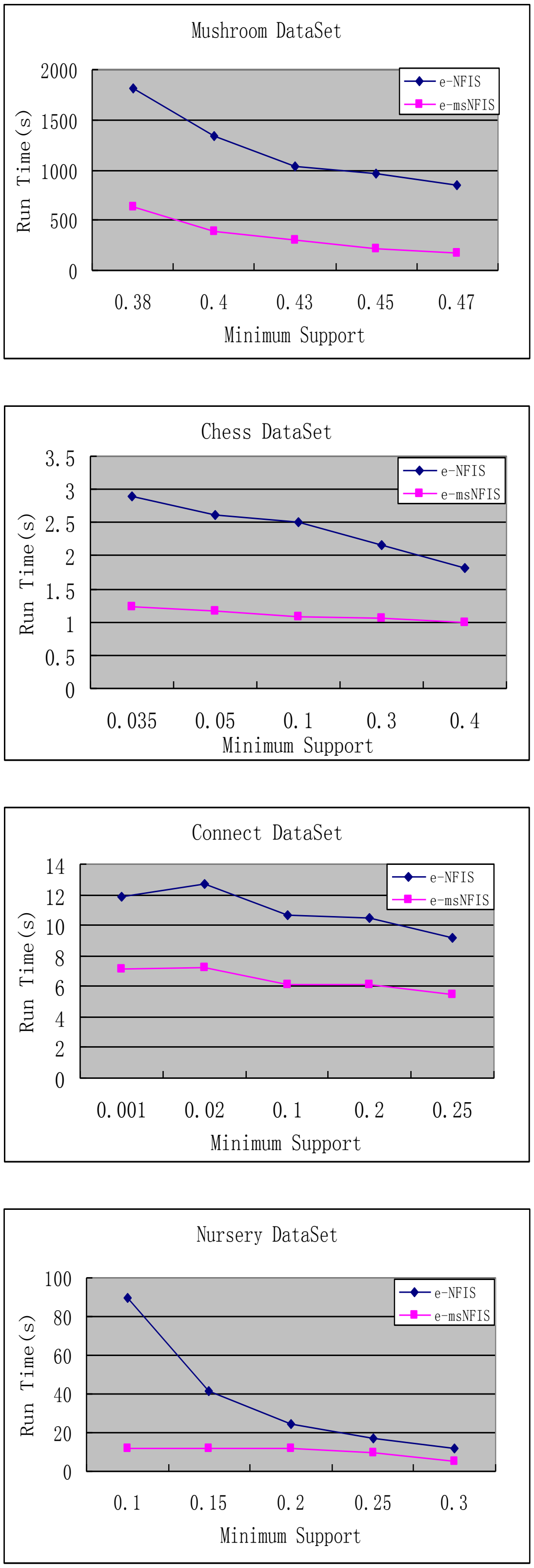

Fig. (1). Run Time(s).

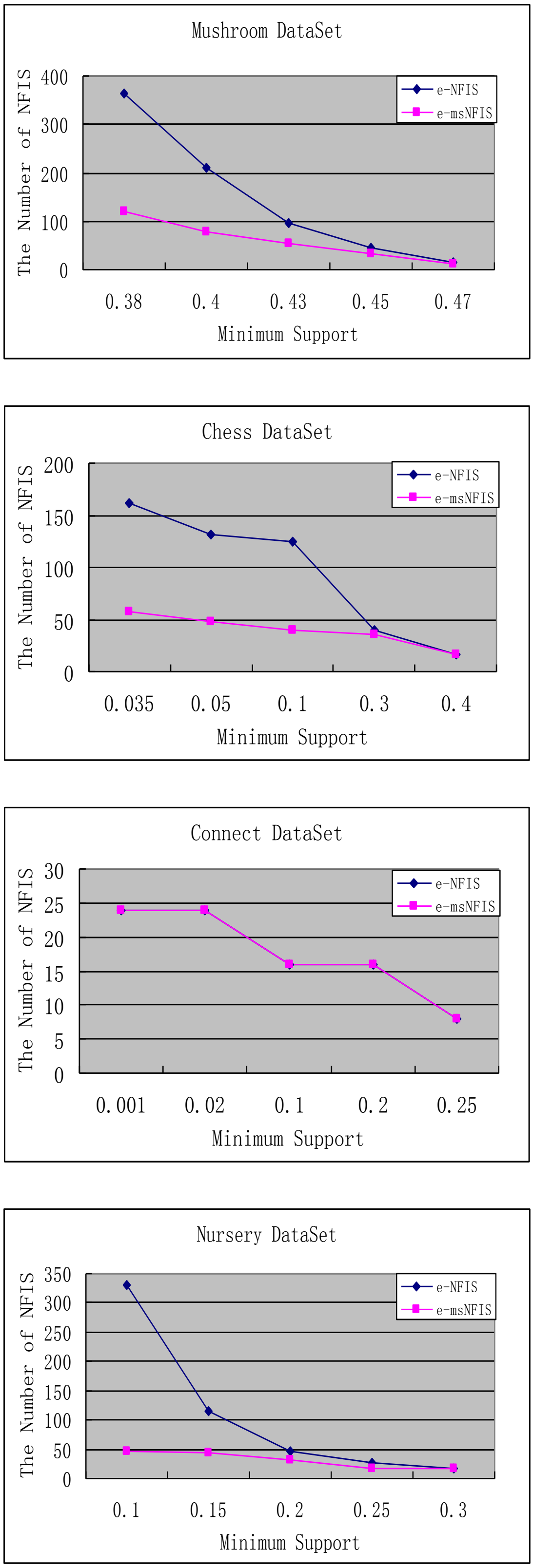

Fig. (2). The Number of NFIS.

From Fig. (**1**), we can see that the gap of two algorithms' run-time is slowly reduced by our method when minimum support user-specified gradually becomes larger. However, the run-time of *e-msNFIS* is less than *e-NFIS*. This is because after running the first step of the algorithm, the number of PFIS in *e-msNFIS* algorithm is less than *e-NFIS* and negative candidate itemsets (NCIS) are generated based on PFIS. The greater the NCIS's number, the longer the runtime. Fig. (**2**) shows that although different itemsets have different number of items, with the increment of minimum support, two algorithms get closer and closer in the number of NFIS. For each dataset, as the minimum support increases, the number of negative frequent itemsets become smaller. In some special circumstances, two algorithms have the same results. (For example, on the connect dataset). This mainly depends on the dense degree of dataset.

6. CONCLUSIONS AND FUTURE WORK

Mining NFIS is much more informative than PFIS in some applications as NFIS often involves non-occurring but interesting item(s). In very few research outcomes reported on NFIS mining, most of the existing techniques only use single minimum support. The single-minimum-support does not reflect the nature of the items and their varied frequencies in the database. To solve this problem, some existing algorithms, such as MSapriori, CFP-growth and MSB_apriori+ have been proposed to mine frequent itemsets with MMS. These methods allow users to assign different minimum supports to different items. But these methods only mine PFIS with MMS. So in this paper, based on *e-NFIS* and MSapriori, we have proposed a method called *e-msNFIS* to mine NFIS with MMS. Furthermore, we have also proposed a method to set up the minimum support to an itemset with negative item(s). Experimental results show that *e-msNFIS* is very effective and efficient.

For future work, we will look for a method to mine NARs from these identified NFIS.

CONFLICT OF INTEREST

The author confirms that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

This work was supported partly by National Natural Science Foundation of China (71271125), Natural Science Foundation of Shandong Province, China (ZR2011FM028) and Scientific Research Development Plan Project of Shandong Provincial Education Department (J12LN10).

REFERENCES

[1] X.-D. Wu, C.-Q. Zhang, and S.-C. Zhang, "Efficient Mining of both Positive and Negative Association Rules," ACM Trans. Information Systems, vol. 22, pp. 381-405, Jul. 2004.

[2] M.L. Antonie, and O. Zaiane, "Mining Positive and Negative Association Rules: An Approach for Confined Rules," Lecture Notes in Computer Science, vol. 3202, pp. 27-38, Sept. 2004.

[3] X.-J. Dong , Z.-D. Niu, X.-L. Shi, X.-D. Zhang, and D.-H. Zhu, "Mining Both Positive and Negative Association Rules from Frequent and Infrequent Itemsets," Lecture Notes in Computer Science, vol. 4632, pp. 122-133, Aug. 2007.

[4] Y.-F. Zhang, C. Wang, and Z.-Y. Xiong, "Improved algorithm of mining association rules with negative items," Computer Engineering and Applications, vol.44, pp.169-171, Sept. 2008.

[5] Y.-F. Zhang, Z.-Y. Xiong, C. Wang, and C.-Y. Liu, "Study on association rules with negative items based on bit string," Control and Decision, vol. 25, pp.37-42, Jun. 2010.

[6] X.-J. Dong, L. Ma, and X.-Q. Han, "E-NFIS: Efficient negative frequent itemsets mining only based on positive ones," in Proc. 3rd International Conf. Communication Software and Networks, Xi'an, 2011, pp.517-519.

[7] B.-G. Yuan, and L. C, "A novel mining algorithm for negative association rules," WRI Global Congress on Intelligent Systems, vol. 2, pp. 553-556, May 2009.

[8] B. Liu, W. Hsu, and Y.-M. Ma, "Mining association rules with multiple minimum supports," in Proc. 5th ACM SIGKDD International Conf. Knowledge Discovery and Data Mining, San Diego, 1999.

[9] Y.-H. Hu, and Y.-L. Chen, "Mining association rules with multiple minimum supports: a new mining algorithm and a support tuning mechanism," Decision Support Systems, vol. 42, pp. 1-24, Oct. 2006.

[10] R. U. Kiran, and P. K. Reddy, "Novel techniques to reduce search space in multiple minimum supports-based frequent pattern mining algorithms," in Proc. 14th International Conf. Extending Database Technology, Uppsala, 2011, pp.11-20.

[11] T.-T. Xu, and X.-J. Dong, "Mining Frequent Patterns with Multiple Minimum Supports using Basic Apriori," in Proc. 9th International Conf. Natural Computation, Shenyang, 2013, pp. 957-961.

[12] X.-J. Dong, Z.-Y. Zheng, Z.-D. Niu, and Q.-T. Jia, "Mining Infrequent Itemsets based on Multiple Level Minimum Supports," in Proc. 2nd International Conf. Innovative Computing, Information and Control, Kumamoto, 2007.

[13] W.-M. Ouyang, and Q.-H. Huang, "Mining direct and indirect association patterns with multiple minimum supports," International Conf. Computational Intelligence and Software Engineering, Wuhan, 2010.

[14] L.-M. Tsai, S.-J. Lin, and D.-L. Yang, "Efficient mining of generalized negative association rules," IEEE International Conf. Granular Computing, San Jose, 2010, pp.471-476.

[15] Z.-X. Ma, and Y.-C. Lu, "Exploding number of frequent itemsets in the mining of negative association rules," Journal of Tsinghua University, vol.47, pp.1212-1215, Jul. 2007.

[16] X.-J. Dong, S.-J. Wang, H.-T. Song, and Y.-C. Lu, "Study on Negative Association Rules," Transactions of Beijing Institute of Technology, vol. 24, pp.978-981, Nov. 2004.

[17] C. Cornelis, P. Yan, X. Zhang, and G.-Q. Chen, "Mining Positive and Negative Association Rules from Large Databases," IEEE Conf. Cybernetics and Intelligent Systems, Bangkok , 2006, pp. 613-618.

[18] X.-D. Wu, C.-Q. Zhang,and S.-C. Zhang, "Efficient Mining of Both Positive and Negative Association Rules," ACM Transactions on Information Systems, vol. 22, pp. 381-405, Jul. 2004.

[19] Y.-C. Lee, T.-P. Hong, and W.-Y. Lin, "Mining association rules with multiple minimum supports using maximum constraints," International Journal of Approximate Reasoning, vol. 40, pp. 44-54, Jul. 2005.

[20] H.-R. Wu, F.-X. Zhang, C.-J. Zhao, "Algorithm for mining association rules with multiple minimum supports," Journal of Harbin Institute of Technology, vol. 40, pp. 1447-1451, Sept. 2008.

Received: May 26, 2015 Revised: July 14, 2015 Accepted: August 10, 2015