2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)
Amity University, Noida, India. June 4-5, 2020

An Approach to Mine Frequent Item Sets Considering Negative Item Values

Reshu Agarwal
Amity Institute of Information Technology
Amity University
Noida, India
ragarwal1@amity.edu

Arti Gautam
Department of Computer Science
G L Bajaj Institute of Technology and Management
Greater Noida, India
arti.gautam@rediffmail.com

Prakhar Dixit
Department of Computer Science
G L Bajaj Institute of Technology and Management
Greater Noida, India
dixitprakhar98@gmail.com

Ajay Rana
AIIT, Amity University Uttar Pradesh
Noida, India
ajay_rana@amity.edu

Abstract— **High utility item set mining identifies the mining item set whose utility satisfies a given limit. Some studies rated the item's values as positive. However, an item set may have negative item values in some applications. Thus, the revelation of high utility item sets with negative item values is required for proper decision-making. This paper proposes a quantitative approach considering negative item values for mining high utility items. An example set out to explain the approach.**

Keywords— Cross-selling; Utility Mining; Association rule mining; Apriori algorithm; Data Mining

I. INTRODUCTION

Association rules and apriori method are widely used for large databases in data mining. The law of association has two stages. In the first step, large item-sets can be found in large databases. Then the union rule is generated. There are several ways to find consistent item sets for traditional databases. However, the item set instance is not sufficient, as it is just one transaction in a database with item sets. Furthermore, the usefulness of the item set cannot be determined from it. It is more beneficial to use utility of an item set to measure the profit of an item set. Now, we consider an example, to understand the meaning of the term utility. In a superstore, consider the gain of the camera is 3000 INR and that of memory card is 15 INR. Now, memory card and camera appears in 10 and 3 transactions respectively. The total gain of the memory card is 150 INR and the total gain of the camera is 9000 INR. Mining memory cards have high repetitions according to repeated item sets. However, the overall advantage of the camera is much more notable than the memory card. Subsequently, mining traditional item sets may not yield the most beneficial item sets. Generally, frequent item set mining does not considers the profit of an item, which is the most important factor [1]. The utility of the item set is denoted by U (a) and is equal to the sum of the utilities of the item set. The utility of the item set by U. U (a) is the sum of the utilities of the item set containing item a. Out of all transactions set, an A set can have a higher utility item set if U (A) is greater than the given threshold. Utility mining is used to find high utility item sets [2].

An idea was proposed in mining the high utility item set [3]. Their proposed work places more attention on the top K high utility closed patterns for top K purpose-guided data mining. Highly desirable statistical patterns are captured with the help of utility. He introduced an algorithm for level-wise item set mining. A new pruning strategy is developed on the basis of utilities to allow pruning of an object with less utility. Utility mining has been developed of late to address the constraint of frequent item set mining, taking into account the customer's desire or purpose and raw information, with item set share structure, weighted item set mining, and object oriented utility can be classified into utility mining[4][5]. In addition, an algorithm based on the conditional tree for mining was proposed by [6]. This novel conditional high utility tree (CHUT) to diminish the search space compressed the database communicated in two steps. A new algorithm called HU-Mine is proposed. In addition, a fast high-utility object set mining algorithm [7] is proposed. This algorithm was divided into two-phases so that a complete set of sets of high utility items can be obtained by efficiently reducing the number of candidates. The first step involves quick identification of candidates using a model implementing "transaction-weighted bottom-off property" at the search space. In the second phase, the high utility item-sets are identified. Further, an efficient algorithm for dynamic databases was proposed for mining high utility item sets [8]. One master node and two slave nodes are used in this system. A slave node was used to calculate the occurrence of each item and each slave node was partitioned for computation for the database. The local table of the slave node was used to store these data. These tables were then sent to the master node. A global table was used to store these data in the master node. Promising and unproven item sets was calculated based on the minimum utility threshold value. A two-stage algorithm was proposed for mining sequential patterns [9]. The first phase is the sort step that is used to sort the database as the key with the customer ID and the minor key at the time of the transaction. The original database is converted into a database of customer sequences. The latter step is the L-item set step to find the set of all L-item sets. An efficient fast update algorithm for computing FUP was proposed. Large item sets in the updated database [10]. They have shown reuse of information from older large item sets. Candidate item sets can be sorted to find new large item sets. He also discussed some optimization techniques to reduce database size during the update process. To further cut the cost, a new algorithm was proposed [11][12]. It has also been observed that sometimes a high utility item set may contain low utility items. The apriori method was used for such item sets. This method was invented to find consistent item sets in the database. The apriori method has two stages. The first stage consists of candidate construction. The second phase involves testing a group of candidates against the data. Traditional association rule mining approaches actually does not reflect the utility of an item set. There are lot of research work on utility mining, but they all failed to consider

978-1-7281-7016-9/20/$31.00 ©2020 IEEE

negative items [13][14]15]. To overcome this problem, a new concept of utility mining was introduced. In this new concept, simplification of utility calculations and size of candidate sets are major challenges to be considered. The next challenge is to search for high utility item sets considering negative values.

This paper focuses on finding high utility item sets considering negative values. An algorithm is proposed to efficiently and effectively detect high utility item sets with negative item values from large databases. The idea behind this algorithm is based on the two-stage algorithm proposed by [16][17]. Using this algorithm, the execution time can be reduced to produce higher utility item values.

II. PROPOSED WORK

The proposed algorithm works in two steps:

Step 1: TWUI method: This method manages the transaction database for the production of TWUI.

Step 2: Filter method: The method manages the negative item move and the high utility item set with negative item values from huge databases.

For mining high utility item sets, filter technique is used. The algorithm works in five steps:

Step1: Input the database.

Step 2: Find high transaction-weighted usage 1-items.

Step3: Transaction-weighted usage use the high transaction-weighted item set to generate a candidate item set.

twu = transaction-weighted usage item set without negative item values.

htwu = high transaction-weighted usage sets i-item without negative item values.

Step4: Search the database to find high transaction-weighted usage item sets.

Step 5: When all htwu are identified, do a final scan of the database. It should be observed that if the value of each item in the item set is negative, it would not be considered a high utility item set. The value of at least one item in the item set must be positive otherwise the item set does not have to examine the database.

Step 6: Finally, items with high utility with negative item values satisfy the set constraint

$$1.htwu \geq threshold.$$

III. NUMERICAL EXAMPLE

Let us consider the utility limit as 60. The sales volume of each item is shown in Table 1 (a). The utility of each item is given in Table 1 (b). Consider the list transaction TID set: {T1, T2, T3, T4, T5, T6, T7, T8, T9, T10} as shown in Table 1. Each row in Table 1 can be taken as an inventory transaction. For example, u ({P, S}) = u ({P, S}, T3) + u ({P, S}, T8) + u ({P, S}, T10) = (1 * 4 + 1 * 4) + (3 * 4 + 5 * 4) + (1 * 4 + 2 * 4) = 60. As in, the utility threshold 60 {P, S} is the higher utility item set. Sometimes a low utility item set can be found in a high utility item set. P is a low utility item as U (P) = 44 <60, but {P, S} is a high utility item set. Be sure not to lose any high utility item sets, so processing of all combinations of high utility item sets should be considered.

The utility calculation is simplified and a two-step algorithm [5] proposes the candidate item set sorting. In the first stage overstimulation of the low utility item set is performed. The transaction utility Ti of any transaction is defined as the sum of utilities of all items in Ti and is denoted by u(Ti) as shown in Table 2. The transaction-weighted usage of an item set A is defined as the amount of transaction utilities. All transactions with item A and are represented by twu(a). For instance, twu(P) = tu (T1) + tu (T3) + tu (T4) + tu (T7) + tu (T8) + tu (T10) = 12 + 12 + P + 16 + 32 + 20 and twu({P, V}) = tu (T1) + tu (T7) + tu (T10) = 12 + 14 + 20 = 46. Actually u (P) = u ({P}, T1) + u ({P}, T3) + u ({P}, T4) + u ({P}, T7) + u ({P}, T8) + u ({P}, T10) = 12 + 12 + 12 = 36 .

Thus, the step proves that it reduces item sets with less utility but never underestimates any item sets. Table 2 gives the transaction utility of each transaction in Table 1. To filter the overestimate of the item in step 2, the database is scanned. For example twu(p) = 98> 60 but u (p) = 48 <60. Item P is seen as having its utility, which is lower than the threshold value 60 to 48. In the same way, all high utility item sets are found. This algorithm is not applicable to databases with negative values because some high utility item sets may be lost. The proposed algorithm is based on a two-stage algorithm. This removes negatively priced items from transactions in large databases. For each transaction in Table 1, Table 3 gives the utility of transactions without negatives item value. Table 4 and 5 explains the algorithm in a better way. The utility limit has been set as 60 for a given 10 transactions. The mining problem is divided into two steps:

1. Transaction Weighted Usage Item Set (TWUI) arises from the mining of the transaction database.
2. High utility item sets are generated by filtering negative item sets from large databases.

The candidate 1 item set C1 is generated from Table 1 by sequentially scanning each transaction. If any item is below the set threshold value, it is removed. In Table 4, all item sets in the candidate 1 item set are above the threshold value so no item set is removed. Items P, Q, R, S, V are written in bold face because they are high transaction weighted usage 1 item set. Further, combining negative value items Q and R with other items resulted in higher utility item sets. Candidate 2 item sets are generated by using high transaction weighted utilization 1 item set as {PQ, PR, PS, PV, QR, QS, QV, RS, RV, SV}. Items PS, QR, QS, RS are written in bold face because they are high transaction weighted usage 2 item set. The item set {QRS} is generated as the candidate 3-item set by the higher transaction weighted usage 2-item set. {P, Q, R, S, V, PS, QR, QS, RS, QRS} are high transaction weighted usage candidate item sets as shown in Table 5. If each item in the item has a negative value, it will not be a high utility item set. For example, items Q, R, QR are removed. Therefore, the high transaction weighted usage set {P, S, V, PS, QS, RS, QRS} became the item set after the item was set. Performing database scans {S} and {PS} results in a higher utility item set as U (S) = 96> 60 and U ({P, S}) = 60 as shown in Table 5.

TABLE I. A TRANSACTION DATABASE AND ITS UTILITY TABLE

TABLE I (A) TRANSACTION TABLE

TID	ITEMS				
	P	Q	R	S	V
T1	1	0	0	0	1
T2	0	1	1	4	0
T3	3	0	0	1	0
T4	2	0	0	0	0
T5	0	1	1	0	2
T6	0	1	2	6	0
T7	1	0	0	0	1
T8	3	0	0	5	0
T9	0	1	2	6	0
T10	1	0	0	2	1

TABLE I (B) UTILITY TABLE

Items	Value per unit
P	4
Q	-5
R	-3
S	4
V	8

TABLE II. TRANSACTION UTILITY

TID	Transaction utility	TID	Transaction utility
T1	12	T6	13
T2	8	T7	14
T3	12	T8	32
T4	12	T9	13
T5	8	T10	20

TABLE III. TRANSACTION UTILITY WITHOUT NEGATIVE ITEM VALUES

TID	Transaction utility without negative item values	TID	Transaction utility without negative item values
T1	12	T6	24
T2	16	T7	14
T3	12	T8	32
T4	8	T9	24
T5	16	T10	20

TABLE IV. CANDIDATE ITEM SETS GENERATED

C1	Transaction weighted utility	C2	Transaction weighted utility	C3	Transaction weighted utility
P	98	PQ	0	**QRS**	64
Q	80	PR	0		
R	80	**PS**	64		
S	128	PV	46		
V	62	**QR**	80		
		QS	64		
		QV	16		
		RS	64		
		RV	16		
		SV	20		

TABLE V. HIGH UTILITY ITEM SETS GENERATED

High transaction weighted utilization candidate item sets		Candidates	Utility
P	**Prune negative item set**	P	44
Q		**S**	96
R		V	40
S		**PS**	60
V		QS	49
PS		RS	49
QR		QRS	34
QS			
RS			
QRS			

The main part of this paper is that it can successfully distinguish high utility item sets with negative item values in short execution time.

IV. CONCLUSION AND FUTURE SCOPE

In this paper, we present a novel algorithm for high utility item sets in databases considering negative item-set. The algorithm can efficiently identify high utility item sets with negative values. In addition, in the future we can derive a sequential pattern from the database to detect this search that includes negative utility values of the items.

REFERENCES

[1] Agarwal, R., lmielinski, T., & Swami, A. (1993). Mining Association rules between set of items in large databases, Proceedings of ACM SIGMOD International Conference on Management of Data (pp.207-216). Washington.

[2] Zaki, M. J., Parthasarthy, S., Ogihara, M., & Li, W. (1997). New Algorithms for fast discovery of Association Rules, Proceedings of KDD (pp.283-286). NY, USA.

[3] Geng, L. & Hamilton, H. J. (2006). Interestingness measures for data mining. ACM Computing Surveys, 38(3), Article 9.

[4] Ramaraju, C., &Savarimuthu N. (2011). A conditional tree based novel algorithm for high utility itemset mining, Proceedings of International Conference on Recent Trends in Information Technology(pp. 701-706).Chennai, Tamil Nadu, India

[5] Liu, Y., Liao, W. k., Choudhary, A. (2005). A Fast High Utility Itemsets Mining Algorithm. Proceedings of the 1st international workshop on Utility-based data mining(pp. 90-99). NY, USA.

[6] Asha, P., Jebarajan, T., &Saranya, G. (2014).A Survey on Efficient Incremental Algorithm for Mining High Utility Itemsets in Distributed and Dynamic Database, International Journal of Emerging Technology and Advanced Engineering, 4(1), 146-149.

[7] Agrawal, R. &Srikant, R. (1995). Mining Sequential Patterns, Proceedings of the Eleventh International Conference on Data Engineering(pp. 3-14).Washington, DC, USA.

[8] Cheung, D. W., Han, J., Ng, V. T., & Wong, C. Y. (1996). Maintenance of discovered association rules in large databases: An incremental update technique. Proceedings of the Twelfth International Conference on Data Engineering (pp.106-114). Washington, DC, USA.

[9] Cheung, D. W., Lee, S. D., & Kao, B. (1997). A general incremental technique for maintaining dis-covered association rules, Proceedings

of the Fifth International Conference on Database Systems for Advanced Applications (pp. 185-194).

[10] Sivamathi, C. & Vijayarani, S. (2016). Utility Mining algorithms – A Comparative Study. Journal of Applied information Science, 4(1), 38-45.

[11] Vijayarani, S., Sivamathi, C., & Suhashini, N. (2016). An Enhanced HUI Miner Algorithm to Retrieve Optimum Number of High Utility Itemsets. International Journal for Scientific Research & Development, 4(18), 74-78.

[12] Hilderman, R. J., Carter, C. L., Hamilton, H. J., &Cercone, N. (1998). Mining market basket data using share measures and characterized itemsets. Proceedings of the Second Pacific-Asia Conference on Research and Development in Knowledge Discovery and Data Mining (pp.72–86). London, UK.

[13] Tseng, V. S., Chu, C. J., & Liang, T. (2006). Efficient mining of temporal high utility itemsets from data streams. *Proceedings of ACM KDD workshop on utility-based data mining*, pp. 1105–1117, New York, USA.

[14] Yao, H., Hamilton, H. J., & Butz, C. J. (2004). A foundation approach to mining item set utilities from databases. *Proceedings of the 3rd SIAM International Conference on Data Mining*, pp.482-486, Orlando, Florida.

[15] Yao, H., Hamilton, H. J., & Geng, L. (2006). A unified framework for utility-based measures for mining itemsets. *Proceedings of ACM SIGKDD 2nd Workshop on Utility-Based Data Mining* , pp. 28-37, USA.

[16] Chan, R., Yang, Q. &Shen, Y. (2003). Mining high utility itemsets, Proceedings of the Third IEEE International Conference on Data Mining(pp. 19-26). Washington, DC, USA.

[17] Agarwal, R. and Mittal, M. (2019). Inventory Classification using Multi-level Association rule mining, International Journal of Decision Support System Technology, 11(2), pp. 1-12. IGI Global.