TRANSACTION PROFITABILITY USING HURI ALGORITHM [TPHURI]

Jyothi Pillai[1] and O.P.Vyas[2]

[1]Associate Professor, Bhilai Institute of Technology, Durg, Chhattisgarh, India
[2]Professor, Indian Institute of Information Technology Allahabad, U.P., India

ABSTRACT

Business intelligence (BI) is formulation of business strategies which help organizations to achieve its objectives and to predict its future. Data mining is often referred as BI in the domain of business. One of the major tasks in data mining is Association Rule Mining (ARM). ARM techniques incorporated in BI systems can be utilized in business decision-making such as retail shelf management, catalog design, customer segmentation, cross-selling, quality improvement and bundling products marketing.

ARM technique is used for the identification of frequent itemsets from huge databases and then generating strong association rules by considering each item having same value. But in a large number of real world applications, items have different values according to their impact on the respective decision making processes. Traditional ARM techniques cannot fulfil the arising demands from these applications. The data mining researchers are continuously improving the quality of ARM technique by incorporating the utility of items. The utility of item is decided by its contribution towards the business profit or quantity of the item sold, etc. Hence Utility mining focuses on identifying the itemsets with high utilities.

Jyothi et al proposed HURI algorithm in [2] for producing high utility rare itemset according to users' interest. An algorithm ***Transaction Profitability using HURI [TPHURI]*** *is proposed in this paper which is a modified version of HURI. TPHURI finds profitable transactions consisting of high utility rare items and also finds the share of such items in the overall profit of the transactions.*

KEYWORDS

Business Intelligence, Association Rule Mining, Utility, Rare Itemset

1. INTRODUCTION

Association rule mining (ARM) algorithms are used to find frequent itemsets and then generate association rules. Association rule mining (ARM) algorithms are used to find frequent itemsets and then generate association rules. In many real world applications such as medical, security, business, etc, items have got different values according to user's perspective. Thus Utility Mining considers different utility values for different items.

Utility Mining is helpful in finding the utility which is a measure to find the usefulness or profitability of an itemset. The total utility of an itemset depends on internal utility and external utility. The internal utility or local transaction utility of an itemset is obtained from the transactional information such as the total quantity of the itemset in a particular transaction. The external utility of an itemset is obtained also from external information sources other than from transactions such as the contribution of itemset towards business profit. The external utility is

generally assigned to itemsets by considering the user's preferences. The goal of utility mining is to identify high utility itemsets which drive a large portion of the total utility [8].

Classical ARM algorithms consider the presence of an item in a transaction to be more important than its absence. The patterns that are rarely found in a database are known as infrequent patterns. The itemset whose support value is less than the maximum support threshold is defined as infrequent itemset or rare itemset [5]. In many business applications, rare itemsets may bring some unexpected acceptable profits. For expanding the business, the supermarkets or stores may shortlist high profitable rare items and then their quantity can be increased for increasing the business profitability.

In this paper, it is proposed that **Transaction Profitability using HURI [TPHURI]** finds profitable transactions consisting of high utility rare items from transactional dataset. TPHURI is used for Transaction Utility Mining to find the share of such items in the overall profit of the transaction dataset. Profitable or interesting transactions are those in which customers purchase high utility rare itemsets. The outcome of TPHURI would enable the top management or business analyst in crucial decision-making such as catalog design, providing credit facility, cross marketing, finalizing discount policy, analyzing consumers' buying behaviour, organizing shelf space, loss-leader analysis and quality improvement in supermarket [3].

The rest of paper is organized as follows. Section 2 presents some related works, Section 3 discusses theoretical definitions and section 4 proposes TPHURI algorithm; an application algorithm using HURI algorithm. Section 5 presents conclusion and future work.

2. LITERATURE SURVEY

Unlike traditional ARM algorithms, the main aim of utility mining algorithms is to discover itemsets having high utilities. For addressing the limitations of AMR technique, Yao et al defined utility mining model [8] for generating high utility itemsets.Yao et al proposed two more utility-based itemset mining algorithms UMining and UMining_H for generating all high utility itemsets by quantifying user preferences [6].

In utility mining downward closure property is not applicable, hence lots of time is consumed in generation of candidate itemsets. In [8], Ying Liu et al presented a Two-Phase algorithm to generate high utility itemsets efficiently by pruning down the number of candidates. In the first phase, the *transaction-weighted utilization mining* model is proposed which applies *Transaction-weighted Downward Closure Property* on the search space to expedite the identification of candidates [8]. High utility itemsets are identified in the second phase by performing one extra database scan.

In [6], Saravanabhavan et al presented a novel utility frequent-pattern efficient tree structure for mining high utility itemsets. For mining utility patterns the authors have used the pattern growth methodology. The authors claim that efficiency of high utility itemsets mining is improved by using two major concepts: 1) the large database is compressed into smaller data structures and also the repetition of the database scans can be avoided by using utility FP-tree; 2) by utilizing the pattern growth method the search space can be reduced by avoiding generation of a large number of candidate sets.

A new algorithm, named Rarity, is presented by Luigi T et al for mining rare itemsets discovered from large databases [4]. A High Utility Rare Itemset Mining [HURI] algorithm is proposed by Jyothi et al for finding high utility rare itemsets according to users' preferences [6]. Using HURI algorithm [6], high utility rare itemsets are generated in two phases:-

(i) In first phase, rare itemsets are generated by considering those itemsets which have support value less than the maximum support threshold.
(ii) In second phase, by inputting the utility threshold value according to users' interest, rare itemsets having utility value greater than the minimum utility threshold are generated.

3. PROBLEM DEFINITION

In this section related theoretical concepts of the proposed algorithm TPHURI are presented [3].

DEFINITION 3.1(TRANSACTIONAL DATASET) A transactional dataset is a collection of transactions where each transaction is a record of items [3]. Let I be a set of quantities of items I={i_1, i_2, i_3,… , i_m}and D be a set of transactions {T1,T2,…,Tn} with items, where each item $i \in I$ (table 1). Each transaction in D is assigned a transaction identifier (T_ID).

For eg. Table 1 is a transactional dataset D consisting of 25 transactions and 20 items.

DEFINITION 3.2 (INTERNAL UTILITY) The internal utility value of item i_p in a transaction Tq, denoted o(i_p, Tq) is the value of an item ip in a transaction Tq (Table 2**)**. The internal utility reflects the occurrence of the item in a transaction database [3]. The set of utilities is defined as U={u_1, u_2, u_3,… , u_k} (table 2).

For e.g., in transaction T_{19}, the quantities of items A0001, B0002, C0003, D0004, E0005, F0006, G0007,… are 0,0,0,2,0,4,0,… respectively. Internal utility of item G in transaction T6 is o(G0007, T6) = 0, while internal utility of item G0007 in Transaction dataset D (table 1) is o(G0007, D) = 8.

DEFINITION 3.3 (EXTERNAL UTILITY) The external utility value of an item is a numerical value s(i_p) associated with an item i_p such that s(i_p)=u(i_p), where u is a utility function assigning utility values according to user preferences (table 2) [3].

From table 3, external utility of item G0007 is s(G0007) = u(G0007) = 6.

DEFINITION 3.4 (ITEM UTILITY) The utility of an item i_p in a transaction Tq, denoted U(i_p, Tq) is product of o(i_p, Tq) and s(i_p), where o(i_p, Tq) is the internal utility value of i_p, s(i_p) is the external utility value of i_p**(**table 3**) [3]**.

For e.g. total utility of item A is U(G0007) = s(G0007) * o(G0007) = 6 * 8 = 48 (table 2).

DEFINITION 3.5 (UTILITY TABLE) A utility table UT **(**table 2**)** is a table containing items and their corresponding utility values where each item i has some utility value u_j in U={u_1, u_2,…,u_k} for some k > 0 [3].

DEFINITION 3.6 (TRANSACTION UTILITY) The transaction utility value of a transaction, denoted as U(Tq) is the sum of utility values of all items in a transaction Tq (table 1, table 2). The transaction utility reflects the utility of items in a transaction database [3].

For e.g., the transaction utility of the transaction T1,
U(T1) = U(A0001)+U(B0002)+U(C0003)+U(D0004)+ … + U(T0020) = 37

DEFINITION 3.7 (UTILITY MINING) Utility Mining is used to find those itemsets having utility values greater than user defined minimum utility threshold. The utility of an itemset *X*, i.e., *u*(*X*),

is the sum of the utilities of itemset X in all the transactions containing *X*. An itemset *X* is called a *high utility itemset* if and only if *u*(*X*) >= *min_utility*, where *min_utility* is a user-defined minimum utility threshold [YH2004]. Identification of the itemsets with high utilities is called as Utility Mining [3].

DEFINITION 3.8 (RARE ITEMSET MINING) Rare itemsets are those itemsets which occur infrequently in the transactional dataset. In many practical situations, rare itemsets having high utilities provide very useful insights to the user. Rare patterns may also indicate the occurrence of exceptional situations in the data. For e.g. If {Fire=Yes} is frequent but {Fire=Yes, Alarm=ON} is infrequent, then latter is an interesting infrequent pattern because it may indicate faulty alarm system [3], [4].

Rare itemset mining is a challenging task where the key issues are: -

(i) Identifying interesting rare patterns and
(ii) Efficiently discovering them in large datasets.

4. PROPOSED ALGORITHM

Algorithm TPHURI
Description: Finding High Utility Rare Itemsets of users' interest and Profitable Transactions
Ck: Candidate itemset of size k *Lk*: Rare itemset of size k
For each transaction t in database
do begin
increment support for each item i present in t
End
L1= {Rare 1-itemset with support less than user provided max_sup};
for(*k*= 1; *Lk*!=Ø; *k*++)
do begin
Ck+1= candidates generated from *Lk*;
//loop to calculate total utility of each item
For each transaction t in database
do begin
Calculate total quantity of each item i in t
Find total utility for item i using following formulae:
Total_utility(item i) = internal_utility(item i) * external_utility(item i)
End
//loop to find rare itemsets and their utility
for each transaction *t* in database
do begin
increment the count of all candidates in *Ck+1* that are contained in *t*
Lk+1 = candidates in *Ck+1 less than* max_support
Add Lk+1 to the Itemset_Utility table in database and calculate rare itemset Utility using formulae:
$$\textbf{Utility(R, t)} = \sum_{\text{for each individual item i in R}} \textbf{(u(i, t));}$$
End
// loop to calculate profit of each transaction and then to find profitable transactions
For each transaction t in database
do begin
Set profit of each transaction in transaction_utility table as
Profit_transaction_t = utility(item i) * quantity(item i in t);
If (profit_transaction_t > user_provided_transaction_utility)
Then Transaction is a profitable transaction
Else Transaction is a non-profitable transaction
End
//loop to calculate whole database utility
For each transaction t
do begin
Db_utility = ∑(profit of each transaction in transaction_utility table)
End
//loop to calculate share of each rare itemset in whole database using Db_utility
For each itemset iset in itemset_utility table
do begin
Share[iset] = utility[iset] / Db_utility;
If (share[iset] > user_provided_threshold for high_profitable_rare_itemset)
Then iset is a rare_itemset which is of user interest
Else iset is a rare itemset but is not of user interest
End
Return rare_itemsets of user interest, profitable_transactions
END

Figure 1: Pseudo Code for TPHURI

A High Utility Rare Itemset Mining [HURI] algorithm is proposed by Jyothi et al for generating high utility rare itemsets according to users' interest [2]. HURI can be used in a variety of business applications for increasing the business profitability. Jyothi et al presented a very innovative idea for customer utility mining by using HURI as a base. CSHURI algorithm, **C**ustomer **S**egmentation using **HURI,** presented in [3] finds those customers who purchase high profitable rare items and then classify the customers according to some criteria.

Another application of HURI is presented in this paper. The authors propose an algorithm **Transaction Profitability using HURI [TPHURI]** which finds profitable transactions consisting of high utility rare items and also finds the share of such items in the overall profit of the transactions.

Transaction Profitability using HURI [TPHURI] algorithm uses two-phase HURI algorithm [2] for finding profitable transactions. Profitable or interesting transactions are those in which customers purchase high utility rare itemsets.

TPHURI algorithm consists of following three phases:-

(i) In first phase, rare itemsets are generated from data set having support value less than the maximum support threshold. Rare rules are those rules appearing below the maximum support value.
By setting the value of maximum support threshold to 40%, the rare itemsets generated from table 1are listed in table 4.

(ii) In second phase, high utility rare itemsets having utility value greater than the transaction utility threshold are generated.
If high utility threshold is set as 80, the high utility rare itemsets generated are listed in table 5.

(iii) At last in the final phase, by setting the transaction utility threshold, profitable transactions consisting of high utility rare items are found.
For e.g.; by setting the user provided transaction utility as 45, transactions can be classified as Interesting (Profit transaction > 45) or Uninteresting (table 6).

Also the share of high utility rare items in the overall profit of the transactions is found in the last phase. A concept, *itemset share*, is proposed in [1] which can be considered as utility because it reflects the impact of sales of itemset on the itemset cost or profit. Itemset share is defined as a fraction of some numerical value, such as total quantity of items sold or total profit.

The final outcome of TPHURI is a set of profitable transactions consisting of high utility rare itemsets which would enable the top management or business analyst in crucial decision-making. The knowledgeable output generated from TPHURI will be applicable for the following business decision-making processes –

- catalog designing
- allocation of credit facility
- implementing discount policy
- analyzing customers' buying behavior and loyalty
- classification of customers
- customer retention management
- organizing shelf space
- quality improvement of products
- demand forecasting

- monitoring movement of high-rated stock and predicting future stock value
- planning for promotion of high utility products
- Sales up-gradation
- Global marketing trend analysis
- High utility product diversification and pricing

Table 1 A transaction database D

TID	A0001	B0002	C0003	D0004	E0005	F0006	G0007	H0008	I0009	J0010	K0011	L0012	M0013	N0014	O0015	P0016	Q0017	R0018	S0019	T0020
1	1	2	2	0	0	1	1	1	2	0	1	1	5	0	0	1	4	1	1	0
2	0	0	0	0	1	3	0	0	0	4	1	1	0	0	0	1	0	1	1	0
3	0	1	0	1	1	0	1	0	1	0	0	0	1	0	1	0	1	1	0	0
4	1	0	1	0	1	0	1	1	1	1	0	0	1	5	1	0	0	0	0	1
5	0	0	2	0	0	0	0	1	0	0	1	0	0	0	1	0	0	0	1	1
6	0	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
7	2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	4	1	0	0
8	0	0	1	0	2	4	0	2	0	0	0	1	0	0	1	1	0	4	0	1
9	0	0	1	0	2	3	1	0	1	5	0	0	3	2	0	1	5	0	1	1
10	1	0	1	1	1	0	0	0	0	3	1	1	0	4	0	0	0	3	0	1
11	1	0	0	0	0	1	0	0	0	3	1	0	0	0	1	0	0	3	0	1
12	2	0	1	0	0	3	0	1	0	2	0	1	1	1	0	1	0	3	0	1
13	1	1	1	0	1	0	0	0	1	1	1	1	0	0	1	1	0	2	0	1
14	1	0	0	0	0	0	1	0	1	0	0	0	1	5	0	0	0	5	0	1
15	0	0	0	1	2	0	0	1	0	0	0	1	0	2	0	1	1	1	0	1
16	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0
17	1	3	0	0	1	4	0	0	0	0	0	0	5	0	0	0	0	0	1	0
18	2	2	1	1	1	0	1	0	1	1	2	1	4	0	0	0	1	1	0	0
19	0	0	0	2	0	4	0	1	2	0	0	0	2	0	1	0	1	0	1	1
20	0	0	0	2	1	0	0	0	0	0	0	0	1	1	1	0	1	0	1	0
21	1	3	2	0	0	0	0	0	2	0	1	0	3	2	1	1	0	2	0	1
22	0	1	0	0	0	0	0	1	1	1	0	0	1	0	0	1	1	0	0	0
23	0	0	0	0	2	1	1	0	1	0	0	1	1	0	0	0	0	2	1	1
24	0	0	2	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	0
25	1	3	0	1	1	2	1	0	1	2	1	0	0	1	0	0	2	2	0	0

Table 2 Utility table

Item	Internal Utility	External Utility
A0001	15	4
B0002	17	1
C0003	16	3
D0004	9	9
E0005	18	7
F0006	26	5
G0007	8	6
H0008	10	1
I0009	16	1
J0010	23	4
K0011	10	3
L0012	9	1
M0013	31	1
N0014	23	2
O0015	10	3
P0016	10	1
Q0017	21	1
R0018	33	1
S0019	8	7
T0020	13	0

Table 3 Transaction utility of the transaction database

T_id	transaction utility	T_id	transaction utility	T_id	transaction utility	T_id	transaction utility	T_id	transaction utility
1	37	6	16	11	25	16	2	21	46
2	30	7	17	12	28	17	20	22	10
3	23	8	28	13	30	18	46	23	26
4	40	9	58	14	47	19	24	24	16
5	14	10	43	15	25	20	18	25	40

Table 4 Rare Itemset Table

Rare itemsets	List of rare itemsets
1-itemset	{D0004}
	{G0007}
	{L0012}
	{S0019}
2-itemset	{D0004, G0007}
	{D0004, L0012}
	{D0004, S0019}
	{G0007, L0012 }
	{G0007, S0019 }
	{L0012, S0019}
3-itemset	{D0004, G0007, L0012}
	{D0004, G0007, S0019}
	{D0004, L0012, S0019}
	{L0012, G0007, S0019}
4-itemset	{D0004, G0007, S0019, L0012}

Table 5 High Utility Rare Itemset Table

High Utility Rare itemsets	List of high utility rare itemsets
1- itemset	{D0004}
2- itemset	{D0004, G0007}
	{D0004, L0012}
	{D0004, S0019}
	{G0007, S0019 }
3- itemset	{D0004, G0007, L0012}
	{D0004, G0007, S0019}
	{D0004, L0012, S0019}
	{L0012, G0007, S0019}
4- itemset	{D0004, G0007, S0019, L0012}

Table 6 Table showing profitable transactions

Transaction Id	Transaction Utility	Share	Transaction Type
1	37	3.8541666666666...	uninterested
2	30	3.125	uninterested
3	23	2.3958333333333...	uninterested
4	40	4.1666666666666...	Rare(high) utility
5	14	1.4583333333333...	uninterested
6	16	1.6666666666666...	uninterested
7	17	1.7708333333333...	uninterested
8	28	2.9166666666666...	uninterested
9	58	6.0416666666666...	Rare(high) utility
10	43	4.4791666666666...	Rare(high) utility
11	25	2.6041666666666...	uninterested
12	28	2.9166666666666...	uninterested
13	30	3.125	uninterested
14	47	4.8958333333333...	Rare(high) utility
15	25	2.6041666666666...	uninterested
16	2	0.2083333333333...	uninterested
17	20	2.0833333333333...	uninterested
18	46	4.7916666666666...	Rare(high) utility
19	24	2.5	uninterested
20	18	1.875	uninterested
21	46	4.7916666666666...	Rare(high) utility
22	10	1.0416666666666...	uninterested
23	26	2.7083333333333...	uninterested
24	16	1.6666666666666...	uninterested
25	40	4.1666666666666...	Rare(high) utility

5. CONCLUSIONS

Data mining techniques can be used by enterprises for minimizing purchasing costs; ranking suppliers by scoring the quality of supplied goods and services; identifying the effective promotions; identifying profitable or high utility itemsets. After identification of high utility rare itemsets, marketers can do the promotion or advertising of such itemsets to increase the overall profit of the business. **Transaction Profitability using HURI [TPHURI]** algorithm first generates high utility rare itemsets using HURI algorithm. TPHURI then finds profitable

transactions consisting of high utility rare items and also finds the share of such items in the overall profit of the transactions.

The profitability of the companies can be increased by identifying the profitable transactions consisting of high utility itemsets and accordingly marketing strategies can be developed for them. The knowledge generated from TPHURI would aid in crucial business decision-making processes such as catalog design, providing credit facility, cross marketing, finalizing discount policy, analyzing consumers' buying behaviour, organizing shelf space, loss-leader analysis and quality improvement in supermarket. The overall sales will be upgraded by the promotion, diversification and pricing strategies used for the sale of high utility products. TPHURI can be efficiently used in other real time applications such as health-care systems, insurance policies, banking, etc.

REFERENCES

[1] Barber B., Hamilton, H. J. "Extracting share frequent itemsets with infrequent subsets", Data Mining and Knowledge Discovery, 7(2) (2003), pp 153-185.

[2] Jyothi Pillai, O.P.Vyas, "High Utility Rare Itemset Mining (HURI): An approach for extracting highutility rare item sets", i-manager's Journal on Future Engineering and Technology (JFET), ISSN Online: 2230-7184, ISSN Print: 0973 – 2632, Aug.-Oct. 2011.

[3] Jyothi Pillai, O.P.Vyas, "CSHURI – Modified HURI algorithm for Customer Segmentation and Transaction Profitability", International Journal of Computer Science, Engineering and Information Technology (IJCSEIT), Vol.2, No.2, April 2012, pp 79-89.

[4] Luigi T., Giacomo S. and Cosimo B., A Fast Algorithm for Mining Rare Itemsets, Ninth International Conference on Intelligent Systems Design and Applications, 978-0-7695-3872-3/09, ©IEEE Computer Society, DOI 10.1109/ISDA.2009.55, pp1149-1155.

[5] Pang N. T., Michael S. and Vipin Kumar, "Introduction to Data mining", 2009.

[6] Saravanabhavan C. and Parvathi R. M. S., "Utility FP-Tree: An Efficient Approach to Mine Weighted Utility Itemsets", European Journal of Scientific Research,© EuroJournals Publishing, Inc., http://www.eurojournals.com/ejsr.htm, ISSN 1450-216X , Vol.50, No.4, 2011, pp.466-480.

[6] Yao, H., Hamilton, H. J., "Mining itemset utilities from transaction databases", Data and Knowledge Engineering, December 2006, Volume 59, pp 603-626

[7] Yao, H., Hamilton, H. J., and Butz, C. J. "A Foundational Approach to Mining Itemset Utilities from Databases", Proceedings of the 4th SIAM International Conference on Data Mining, Florida, USA, 2004.

[8] Ying Liu, Wei-keng Liao, Alok Choudhary, A Fast High Utility Itemsets Mining Algorithm, UBDM '05 , August 21, 2005, Chicago, Illinois, USA.

Authors

Mrs. Jyothi Pillai is Associate Professor in Department of Computer Applications at Bhilai Institute of Technology, Durg (C.G.), India. She is a post-graduate from Barkatullah University, India. She is a Life member of Indian Society for Technical Education. She has a total teaching experience of 18 years. She has a total of 24 research papers published in National / International Journals / Conferences into her credit. Presently, she is pursuing Ph.D. from Pt. Ravi Shankar Shukla University, Raipur under the guidance of Dr. O.P.Vyas, IIIT, Allahabad.

Dr.O.P.Vyas is currently working as Professor and Incharge Officer (Doctoral Research Section) in Indian Institute of Information Technology-Allahabad (Govt. of India's Center of Excellence in I.T.). Dr.Vyas has done M.Tech.(Computer Science) from IIT Kharagpur and has done Ph.D. work in joint collaboration with Technical University of Kaiserslautern (Germany) and I.I.T.Kharagpur. With more than 25 years of academic experience Dr.Vyas has guided Four Scholars for the successful award of Ph.D. degree and has more than 80 research publications with two books to his credit. His current research interests are Linked Data Mining and Service Oriented Architectures.