E: ISSN No. 2349-9443 RNI No.UPENG/2012/42622 VOL.-8, ISSUE-1,January (Part-1) 2019
Asian Resonance

A Survey on Utility Mining

Abstract

Utility Mining is a research area in data mining. It is mainly focused in market basket analysis to retrieve a high profit from the different combination of itemsets. Utility is the interestingness of user preferences like profit, cost, etc. Utility mining is the enhanced version of association rule mining. In this paper, we give the introduction of association rule mining and its disadvantages, then the importance of utility mining and its basic terminologies, different algorithms and methods which are used in utility mining.

Keywords: Association Rule Mining (ARM), Utility Mining, Support, Confidence, High Utility Itemsets (HUI).

S.J.Vivekanandan
Research Scholar,
Dept. of CSE,
Sathyabama Institute of Science and Technology,
Chennai, India &
Assistant Professor,
Dhanalakshmi College of Engineering,
Chennai, India

G.Gunasekaran
Research Supervisor,
Dept. of CSE,
Sathyabama Institute of Science and Technology,
Chennai, India &
Principal,
J.N.N Institute of Engineering,
Chennai, India

Introduction

From the tons of data, it is important to change the data into useful information, Data mining is used for this purpose. Data mining [1] [2] is used to find unseen information or secret pattern or interesting rules from the large amount of data. Different data mining techniques [3] like classification, clustering, temporal data and so on. Initially, data mining faced with different requirements and challenges [3] to satisfy its need and goals. Data mining is a key role in the process of knowledge discovery. KDD process is the foundation of data mining; it follows the sequence of levels to extract new facts from the data [2]. The different levels are data cleaning, data integration, data selection, data transformation, data mining, pattern evaluation and knowledge representation.

Association Rules Mining (ARM) is one of the most important research areas among the researchers. The objective of ARM is to identify the strong association rules or the relationship between the itemsets from the mass amount of data. Association rules are generally in the form E→F, where E is antecedent and F is consequent. It means, if E occurs, then F also possible to occur. Association rule mining is mainly used in market basket analysis to identify the customer purchase habits and also to increase the profit.

Let $I = \{i_1, i_2, i_3, \ldots, i_n\}$ be a set of items and Tran_DB = $\{t_1, t_2, t_3, \ldots, t_n\}$ a set of transactions where every transaction is also a set of items. Association rules can be measured by two interesting measures such as support and confidence.

Support is the number of times an item available in the transaction. Support (i_1) = Number of times i_1 appears in the database divided by the total number of transactions. Support (i_1i_2) = Number of times i_1i_2 appears together in the database divided by the total number of transactions. Confidence is the probability of A occurs when B also occurs, where A and B are two different itemsets. Confidence ($i_1 \rightarrow i_2$) = ratio of Support (i_1i_2) to the Support (i_1). An item or itemsets are greater than or equal to the minimum support value, then the itemsets are called frequent item or frequent itemsets. An item or itemsets are less than the minimum support value, then the item or itemsets are called infrequent item or infrequent itemsets Association mining is a process of finding strong association rules [4] [5]. It describes in two different steps

Step 1: Frequent Itemsets Generation

Compute all itemsets that are greater than or equal to support value.

Step 2: Association Rule Generation

Based on step 1, pull out all the rules that are greater than or equal to confidence value. Those rules are called strong association rules.

Different algorithms of ARM work on the basis of Apriori property.

Property 1

if an itemset is an infrequent itemset, then all its supersets also infrequent itemset.

Property 2

if an itemset is a frequent itemset, then all its subsets also frequent itemset.

The transactional database can be Boolean transaction, i.e. it contains only 0 or 1 entries. 0 represent an item or itemset has not purchased. 1 represent an item or itemset has purchased.

Disadvantages of ARM

1. Multiple scans of database
2. Lots of candidate generation
3. Rare items /itemsets are neglected
4. Not significance to the user

We discuss one example, Table I denotes profit of each item, Table II denotes a transactional database and Table III represents support and profit of all itemsets.

Table I

Item	Profit
A	5
B	100
C	40

Here, it gives unit profit for each item, so it will be helpful to find the total cost of a transaction and also to find the total cost of an item in the transactional database.

Table II

Transaction TID	Quantity of Items purchased in transaction		
	A	B	C
T101	2	0	1
T102	4	0	2
T103	4	1	0
T104	0	1	1
T105	5	1	2
T106	10	1	5
T107	4	0	2
T108	1	0	0
T109	3	0	0
T110	5	0	0

Here, it contains 10 transactions; each transaction has unique TID and quantity of items purchased. If you apply apriori algorithm [4] with minsup = 40, it generates possible combination of itemsets, i.e. A, B, C, AB, AC, BC, ABC.

Table III

Item / Itemset	Support	Profit
A	90	190
B	40	400
C	60	520
AB	30	395
AC	50	605
BC	30	620
ABC	20	555

Here, it denotes support and their profit of each itemset. In this example, minsup =40 so it takes A, B, C, AC only because it consider only support value i.e. frequent itemsets. Even though AB, BC, ABC are having more profit, it is not taken because it considered as rare itemset. So the objective of market basket analysis is not satisfied, i.e. profitable itemsets are simply missing in this example. So we need a new approach to handle this problem.

Introduction to utility miniing

Utility mining is a new research area in data mining. It is mainly focused in market basket analysis to retrieve a high profit from the different combination of itemsets. Utility is the interestingness of user preferences like profit, cost, etc. Utility mining is the enhanced version of association rule mining. The profit of an itemset depends not only on the support value of the itemset but also on the cost of the items in the itemset. In utility mining, important research area is high utility mining. Itemsets having utility values, which are greater than the minimum utility value, then the itemsets are called high utility itemsets. In the next sections, we discuss the basic terminologies used in high utility mining and various algorithms used in utility mining.

Basic Terminologies of Utility Mining

Support

Support denotes the number of times an item or itemset present in the transaction. In Table II, Support (A) = 9/10, Support (B) = 4/10, Support (C) = 6/10, Support (AC) = 5/10 and Support (BC) = 3/10.

Frequent Itemset

An item or itemsets are greater than or equal to the minimum support value, then the itemsets are called frequent itemsets.

Infrequent Itemset or Rare Itemset

An item or itemsets are lesser than the minimum support value, then the itemsets are called infrequent itemsets or rare itemsets.

Internal Utility (IU)

Internal utility refers the quantity of each item in the transaction. It is used to find the utility of an item as well as the utility of a transaction. In Table II, T101 have (A, 2) and (C, 1) are the internal utilities of the itemsets. Similarly T106 (A, 10), (B, 1) and (C, 5) are the internal utilities of the itemsets

External Utility (EU)

External utility is the profit of each item available in the transaction. In Table I, (A, 5), (B, 100) and (C, 40) denotes its profit associate with it.

The Utility of an item or itemset (U)

The Utility of an item or itemset is the multiplying its internal utility and external utility. It can be represented as U = IU * EU. For example, U (A, T101) = 2 * 5 = 10, U (B, T104) = 1*100 = 100, U(C, T106) = 5 * 40 = 200, U (AC, T107) = 4 * 5 + 2 * 40 = 100, U (ABC, T106) = 10 * 5+1 * 100+5 * 40 = 350.

The Utility of an item or itemset for Database (UD)

The Utility of an item or itemset for the database is the product of the quantity of the item or itemsets purchased in the database and its corresponding external utility (profit). For example, UD (A) = 38 * 5 = 190, UD (B) = 4 * 100 = 400 and UD (C) = 13 * 40 = 520.

Transaction Utility (TU)

Transaction utility is computed by multiplying each item's internal utility with its external utility. For

example, TU (T101) = 2 * 5 + 0 * 100 + 1 * 40 = 50, TU (T104) = 0 * 5 + 1 * 100 + 1 * 40 = 140, TU (T105) = 5 * 5 + 1 * 100 + 2 * 40 = 205, TU (T106) = 10 * 5 + 1 * 100 + 5 * 40 = 350.

Transaction-weighted Utility of an itemset (TWU)

Transaction-weighted utility of an itemset Y is defined as the sum of the transaction utilities (TU) of all the transactions that contains the itemset Y. For example, TWU (B) = TU (T103) + TU (T104) + TU (T105) + TU (T106) = 120 + 140 + 205 + 350 = 815. TWU (A) = TU (101) + TU (102) + TU (103) + TU (105) + TU (106) + TU (107) + TU (108) + TU (109) + TU (110) = 50 + 100 + 120 + 205 + 350 + 100 + 5 + 15 + 25 = 970. TWU (C) = TU (101) + TU (102) + TU (104) + TU (105) + TU (106) + TU (107) = 50 + 100 + 140 + 205 + 350 + 100 = 945.

High Utility Itemset (HUI):

The Utility of an itemset is greater than or equal to the minimum utility threshold (User specified value, denoted as min_util), then the itemset is called high utility itemset.

High Transaction Weighted Utility Itemset (HTWUI)

An itemset Y is called a high transaction weighted utility itemset, then its transaction-weighted utility (TWU) must be greater than the minimum threshold utility (min_util).

Transaction-weighted downward closure property

For any itemset Y, if Y is not a HTWUI, then any superset of Y is a low utility itemset. For example TWU (A) = 970, TWU (B) = 815 and TWU (C) = 945 if min_util is 900, then item B is not HTWUI. Then U (AB) = 120 + 125 + 150 = 395 i.e. low utility itemset.

Different Algorithms and Approaches of Utility Mining

A theoretical approach of utility mining was proposed [6], it describes the utility bound property, the support bound and mathematical model of utility mining was designed based on these properties. This model is the basement for many research works in the field of utility mining.

With the help of mining using expected utility [6], they (Y. Liu, W.Liao, Alok) proposed a new concept called transaction-weighted downward closure property [7] (Already we discussed this property in section II 12th definition). They propose a two phase algorithm for efficient pruning and mining high utility itemsets.

The same authors of [6], mathematical model of utility mining were converted into two different algorithms, i.e. utility mining and utility_H algorithms [7]. Here, they used same properties used in [4] along with utility constraints and different pruning strategies.

In [8], an interesting model was introduced, i.e. utility-frequent mining model to identify all itemsets that are greater than the user specified utility and also greater than minimum support threshold. Also it gave a new way of finding the frequency of an itemset, i.e. quasi-utility-frequency. To find quasi-utility frequency itemsets, they proposed apriori like algorithm. By using these two concepts and algorithm, they have proposed bottom-up two phase algorithm and top-down two phase algorithm.

The enhanced version of two phase algorithm [8] was proposed in the name of Fast Utility frequent mining algorithm [9]. The original has disadvantages like more time consuming in candidate generation and calculating its utility. But FUFM is more efficient and faster than the original 2-phase utility-frequent algorithm. It uses a hash tree data structure [9] for pruning the unwanted itemsets.

An enhanced version of the umining algorithm [7] was proposed a algorithm called Fast Utility mining (FUM) algorithm. The original umining algorithm generates lots of candidate itemsets and there was no proper way to handle the redundant itemsets. That problem was efficiently handled by fast utility mining (FUM) algorithm. By combining fast utility frequent mining algorithm (FUFM) [9] and fast utility mining (FUM) [10], they generated a novel approach to categories high utility items as HUHF – High Utility High Frequency, HULF – High Utility Low Frequency, LUHF – Low Utility High Frequency and LULF – Low Utility Low Frequency [11]. This category of itemsets helps in CRM to identify customer purchase habits, customer segmentation and improve their business profit.

In utility mining, a tree like structure was used for mining high utility itemsets with different pruning strategies for candidate itemsets. Utility pattern (UP – tree) tree is a data structure that can be used to store the information of high utility itemsets. By using UP-tree, a new algorithm was proposed called Utility Pattern Growth (UP - Growth) algorithm [12]. It uses some concepts of FP-tree. UP-Growth is better than all other previous algorithms. By continuing the research, a new algorithm was proposed with updated pruning strategies, i.e. UP-Growth^{+} algorithm [13].

Another approach to store the information of high utility itemsets called utility list structure. It contains information of high utility and pruning status of that item. This list helps to propose an algorithm called High Utility Itemset Miner (HUI – Miner) [14]. In this approach, sorting strategies used in TWU, so it was more efficient than state-of-art algorithms.

In an ARM, infrequent itemsets was missed even though itemsets have high utility. In utility mining, high utility itemsets are generated by using min_support value and min_util value. In this approach, rare itemsets may be neglected. So in [15] a new approach to finding high utility rare itemsets called transaction profitability using high utility rare itemset mining (TPHURI). It finds the high utility rare itemsets and found that these itemsets are more significant in the overall profit. So high utility rare itemset mining is an interesting research area inside the utility mining.

An itemset X is closed itemset (closed frequent itemset) only if X is frequent and no immediate superset of X has same support as X. This closed itemset concept is used in high utility mining and derive an itemset called closed^{+} high utility itemset [16]. They propose three different methods along with this closed^{+} high utility itemset. Apriori based algorithm for mining high utility closed^{+} itemset, closed^{+} high utility itemset discovery and derive all

utility itemsets. All this approach is related and depended on each other. These approaches are efficiently handled the redundant high utility itemsets.

Interactive data mining is the way to provide a platform for the users to learn and understand the problem, so that they can give their possible ideas to solve the problem. Incremental data mining is dealing with real time databases which have lots of insert, update and all other operations. These two concepts are combined with high utility itemsets i.e high utility pattern mining (HUP). To handle this kind of situation, a new tree structure based algorithms were proposed [17]. Incremental high utility pattern lexicographic tree, incremental high utility pattern transaction frequency tree and incremental high utility pattern transaction weighted utilization tree [17]. These are helping to implement interactive and increment data mining efficiently in high utility itemsets.

An interesting area of data mining is privacy preserving data mining (PPDM). It is the task of extracting the confidential data and sensitive data from the large database without compromise any privacy protection of the data. Sanitizing process is the removal of sensitive data from the collection of data. Privacy preserving data mining, sanitizing process, ARM and utility mining are combined to find the high utility itemset without compromising privacy. Two different algorithms were proposed [18] hiding high utility item first algorithm and maximum sensitive itemsets conflict first algorithm which are different from normal utility mining.

In high utility itemset mining, when the value of min_util is low, then a huge number of high utility itemsets will be generated which may not be useful to the user. So that it requires a framework to mine important high utility itemset. To handle this, a new framework was proposed, i.e. Top-K high utility itemsets. To implement this framework, two different algorithms were generated, they are mining Top-K utility itemsets and mining Top-K utility itemsets in one phase [19]. TKU algorithm implemented with the help of UP-Tree data structure [12]. TKO algorithm implemented with the help of Utility list data structure [14].

Infrequent itemsets plays a key role in the profit of the users. We discussed some of the strategies to find high utility rare itemsets. In [20], they proposed a framework to find high utility rare itemsets. They found that the itemsets have 0 values (internal utility) in transactions, surely it will present in high utility infrequent itemset.

In [21], an enhanced closed high utility itemset algorithm was proposed closed potential high utility itemset – list algorithm with different sub-routines for effective pruning and mining the high utility itemsets.

Conclusion

This paper gave an overview of Association rule mining and utility mining. Also, it provides different algorithms and approaches used in utility mining. We hope that this paper will give basic knowledge and ideas of various areas of utility mining.

References

1. *G.K.Gupta, "Text Book: Introduction to Data Mining with Case Studies" 3rd edition, pp. 91-151 PHI Learning Pvt. Ltd. 2019*
2. *J Han, M kamber, "Text Book: Data Mining: Concepts and Techniques", 2nd edition, Morgan Kaufmann publisher, An imprint of Elsevier 2006*
3. *M.-S Chen, Han J, " Data Mining: An Overview from a Database Perspective", IEEE transaction on Knowledge and Data Engineering, Vol.8, No. 6 December 1996.*
4. *Agrawal R, Srikant R, " Fast algorithms for mining association rules", Proceedings of 20th International Conf. on Very large Databases, Santiago, Chile, pp 487-499, 1994.*
5. *Agrawal R, Imielinski T., Swami A, "Mining association rules between set of items in large databases", Proceedings of the ACM SIGMOD Intl. Conf. on Management of Data, Washington, D.C.. may 1993, pp 207-216.*
6. *Hong Yao, J. Hamilton, Cory J Butz, "A Foundational Approach to Mining Itemset Utilities from Databases", Proceedings of 4th SIAM Tntl. Conf. on Data mining, 2004, pp 482-486.*
7. *Ying Liu, Wei-keng Liao, Alok Choudhary, "A Fast High Utility Itemsets Mining Algorithm", Workshop on Utility-based data mining 2005.*
8. *Jieh-Shan Yeh, Yu-Chiang Li, Chin-Chen Chang, "Two-Phase Algorithms for a novel Utility-Frequent Mining model", PAKDD Workshop LNAI 4819, pp. 433-444, 2007.*
9. *Vid Podpecan, Nada Lavrae, Ignor kononenko, "A Fast Algorithm for Mining Utility-Frequent Itemsets", Intl. Workshop on Constraint based mining and Learning at ECML/PKDD, pp. 9-20, 2007.*
10. *Shankar, Purusothaman, Jayanthi, Babu, "A Fast Algorithm for Mining High Utility Itemsets", in proceedings of IEEE International Advance Computing Conference, Patiala, india, pp.1459-1464, 2009.*
11. *Shankar, Purusothaman, Kannimuthu, Priya, "A Novel Utility and Frequency based Approach for improving CRM in etaul business", International Journal of Computer Applications, volume 1 – No. 16. 2010.*
12. *Vincent S. Tseng, Cheng-Wei Wu, Bai-En Shie, Philip S, "UP-Growth: An Efficient Algorithm for High Utility Itemset Mining", in proceeding of 16th ACM SIGKDD International Conference on KDD 2010.*
13. *Vincent S. Tseng, Cheng-Wei Wu, Bai-En Shie, Philip S, "Efficient Algorithms for Mining High Utility Itemsets from Transactional Databases", IEEE Transactions on Knowledge and Data Engineering, vol. 25, No. 8, 2013.*
14. *Mengchi Liu, Junfeng Qu, "Mining High Utility Itemsets without Candidate Generation", in proceedings of CIKM' 12 pp. No. 55-64, 2013.*
15. *Jyothi Pillai, O.P.Vyas, "Transaction profitability using HURI algorithm", Intl. journal of business information systems strategies, vol. 2, No. 1, 2013.*
16. *Vincent S. Tseng, Cheng-Wei Wu, Bai-En Shie, Philip S, "Efficient Algorithms for Mining the concise and lossless representation of High*

Utility Itemsets", IEEE Transactions on Knowledge and Data Engineering, vol. 27, No. 3, 2015.

17. *Chowdhury F Ahmed, Syed K Tanbeer, Byeong, Young Lee, "Efficient tree structures for High Utility Pattern Mining in Incremental Databases", IEEE Transactions on Knowledge and Data Engineering, vol. 21 No.12, 2009.*
18. *Jieh-Shan Yeh, Po-Chiang Hsu, "HHUIF and MSICF: Novel algorithms for privacy preserving utility mining", Expert systems with applications 37, pp 4779-4786, 2010.*
19. *Vincent S. Tseng, Cheng-Wei Wu, Bai-En Shie, Philip S, "Efficient Algorithms for Mining Top-K High Utility Itemsets", IEEE Transactions on Knowledge and Data Engineering, vol. 28, No.1, 2016.*
20. *Sunidhi Shrivastava, Punit kumar, "Analysis on high utility infrequent itemsets mining over transaction database", IEEE International conference on recent trends in electronics information communication technology, india, 2016.*
21. *Bay Vo, Loav Nyugen, Trinh D.D, "An Efficient method for Mining closed potential High Utility Itemsets", IEEE Access special section on utility pattern mining: theoretical analytics and applications, vol. 8, pp. 31813 – 81822, 2020.*