2nd International Conference on Advanced Research in Biology, Engineering, Science and Technology (ICARBEST'16)
Organized by
International Journal of Advanced Research in Biology, Engineering, Science and Technology (IJARBEST)
19th February 2016

An Effective Mining the ConciseRepresentation Based on the Odd Ratio Pattern and Distinctive Closed High Utility Itemsets

[1]Immanuel K, [2]E.Manohar, [3]Dr.D.C.Joy Winnie Wise
[1]M.E.(CSE) , [2]Assistant Professor, [3]Head &Professor
[1]Department of Computer Science, [2]Department of Computer Science, [3]Department of Computer Science
[1]k.immanuel1407@gmail.com, [2]manohar2k@ymail.com,[3]joywinniewise@yahoo.com
[1]Francis Xavier Engineering College, [2]Francis Xavier Engineering College, [3]Francis Xavier Engineering College
Tirunelveli, Tamil Nadu

Abstract

Mining closed high utility itemsets(CHUIs) from databases is an essentialtopic in data mining, which refers to the discovery of itemsets to utilities greater than an end-user specific minimum utility threshold min_util. Apriori is an algorithm used for frequent item set mining. Apriori is designed to operate association rule learning over transactional databases. In our existing system is solved the compact and lossless representation of HUIs from closed high utility itemsets, three powerful algorithms are used AprioriHC (Apriori-based algorithm for mining High utility Closed itemsets), AprioriHC-D (AprioriHC algorithm with Dispose of unpromising and isolated items) and CHUD (Closed High Utility Itemset Discovery) to find this representation. In this project, we propose two efficient algorithms are proposed to reduce competition and also high utilities. There are: (1) AprioriHC-D-ORP(AprioriHC-D algorithm with ORP) and (2) AprioriHC-D-DCI(AprioriHC-D algorithm with Distinctive Closed Itemsets) to find Concise representation. In AprioriHC-D-ORP algorithm explores the ratio of the odds that a case has been exposed to the odds that a control has been exposed. In odds ratio patterns, we consider only dataset D, we write D^{pos} for those transactions in D that are denoted as positive, and D^{neg} for those denoted as negative. We can extend our notations and write sup(E,D) and ORP(E,D) respectively. We propose two approaches, first identify all the equivalence classes and their borders in Datasets D, and second obtain a decomposition of the frequent itemsets in Dpos and extract the border of the corresponding to the support level of in D^{pos}. In each element, R_j in the right bound of the closed pattern of an equivalence class. For each R_j, we check also D^{neg} to see if this R_j have good odds ratio. If it does not have good odds ratio, then its entire equivalence class can be discarded. If it has good odds ratio, then we need to consider the corresponding left bound L_j. We need to move L_j right to a more specialized L'_j, so that the requirements on odds ratio are satisfied. In AprioriHC-D-DCI algorithm uses a level-wise algorithm to find all frequent DCIs. We consider that h be the closure operator that structurally characterizes the disjunctive closure of any itemset I. This algorithm uses closure operator h to conclude that regenerates all the list of Frequent Itemsets.

***<u>Keywords</u>*- **Frequent Itemset, Closed High Utility Itemset, Concise Representation, Utility Mining, DataMining**

1. Introduction

DataMining is a complete process of analyzing large amounts of data and picking out the relevant information. It refers to extracting or mining hidden predictive information from large amounts of data. The data sources can include data bank, data warehouse, the network, other information repositories, or data that are streamed into the system dynamically. [4,13]. Association Rule in Data Mining deals with a vital role in the process of mining data for frequent itemsets. Finding frequent patterns called as association, correlation, and causality analysis. Frequent patterns are the patterns that occur frequently in the data set. Patterns can include itemsets, sequences and subsequences. Frequent itemset mining is an interesting topic of data mining that focuses on looking at sequences of actions or events. A frequent itemset refers to a set of items that often occur together in a transactional data set. Example: bread and milk. .It involves the following steps: cleaning and coordinate data from data sources like databases, flat files, pretreatment of selecting and transformation target data, mining the required knowledge and finally estimation and delivery of knowledge. A data mining algorithm is complete if it mines all interesting patterns. It is often impractical and ineffectual for data mining systems to generate all possible patterns. The task for the frequent itemset mining algorithm is then to find the minimum support. One of its trendy applications is market basket analysis, which refers to the discovery of set of items (itemsets) that are frequentlypurchased together by customers. In data mining, association rule learning is a most popular technique to identifythe interesting relationship between the data stored in large database.Condensed representations have been proposed in (Mannila&Toivonen, 1996) as a useful concept for the optimization of typical data mining tasks. It appears as a key concept within the inductive database framework (Imielinski&Mannila, 1996; Boulicaut et al., 1999; de Raedt, 2002) and (Jean-François Boulicaut., 2005) introduces this research domain, its achievements in the context of frequent itemset mining (FIM) from transactional data and its future trends.

In High Utility Itemset Mining, the objective is to determine itemsets that have utility values above a given utility threshold. The utility bound property of any itemset gives an upper bound on the utility value of any itemset. This utility bound property can be used as aexamining measure for pruning itemsets at early stages that are not expected to qualify as high utility itemsets. High utility itemset mining is a challenging task in frequent pattern mining, which has wide applications. Given a transaction database, FIM consists of discovering frequent itemsets. i.e. groups of items (itemsets) occur frequently in transactions. However, an important limitation of FIM is that it consider that each item cannot come more than once in each transaction and that all items have the same importance (weight, unit profit or value). For example, consider a database of customer transactions containing information about the quantities of items in each transaction and each item has a unit profit value. FIM mining algorithms would discard this information and may thus discover many frequent itemsets achieve a low profit and fail to discover less frequent itemsets that achieve a high profit.

The problem of FIM has been redefine as High-Utility Itemset Mining (HUIM) to consider items can appear more than once in each transaction and each item has a weight (e.g. unit profit). The goal of HUIM is to discover itemsets having a high utility (e.g. developing a high profit).The problem of HUIM is widely recognized more difficulty than the problem of FIM. In FIM, the downward-closure property shows that the support of an itemset is anti-monotonic, that is the supersets of an infrequent itemset are uses infrequent and subsets of a frequent itemset are usesfrequent. This property is very energetic to prune the search space. In HUIM, the utility of an itemset is neither monotonic nor anti-monotonic, so that high lower, equal or higher utility [1]. Thus techniques to prune the search space advanced in FIM cannot besearch space advanced in FIM cannot be directly applied in HUIM.

2. Related Work

Alva Erwin et al [1] have planned high utility itemsets mining that extends frequent pattern mining to find itemsets during a dealings info with utility values higher than a given threshold. However, mining high utility itemsets presents a larger challenge than frequent itemset mining, since high utility itemsets lack the anti-monotone property of frequent itemsets. Transaction Weighted Utility (TWU) planned recently by researchers has anti-monotone property, however it's AN overestimate of itemset utility and thus results in a bigger search house. Bai-En Shie et al [2] has mining high utility mobile ordered patterns by integration mobile data processing with utility mining. 2 tree-based ways area unit planned for mining high utility mobile ordered patterns. Cheng Wei dynasty Wu dialect et al. [3] has mining closed+ high utility itemsets that is a compact and lossless illustration of high utility itemsets. We tend to gift AN economical rule referred to as CHUD (Closed+ High Utility itemset Discovery) for mining closed+ high utility itemsets. Further, a way referred to as DAHU (Derive All High Utility itemsets) is planned to recover all high utility itemsets from the set of closed+ high utility itemsets while not accessing the initial info. Ching-Huang Yun [4] has planned a brand new data processing capability for a mobile commerce surroundings. to raised mirror the client usage patterns within the mobile commerce surroundings, we tend to propose AN innovative mining model, referred to as mining mobile ordered patterns, that takes each the moving patterns and get patterns of consumers into thought. Claudio Lucchese [5] have planned afresh ascendable rule for locating closed frequent itemsets, a lossless and condensed illustration of all the frequent itemsets which will be strip-mined from a transactional info. This rule exploits a divide-and-conquer approach and a bitwise vertical illustration of the info and adopts a specific visit and partitioning strategy of the search house supported an inventive theoretical framework, that formalizes the matter of closed itemsets mining very well.

3. General Definitions

Itemset: Set of things that occur along.

Association Rule: Likelihood that individual things area unit purchased along.

X ® Y wherever X ÇY = zero

Support:supp(X) of associate degree itemset X is that the quantitative relation of transactions during which associate degree itemset seems to the entire variety of transactions. The support worth of X with relevance   is outlined because the proportion of transactions within the information that contains the item-set X. within the example information, the itemsetencompasses a support of 1/5=0.2 since it happens in a pair of0% of all transactions (1 out of five transactions). The argument of supp() could be a set of preconditions, and therefore becomes additional restrictive because it grows (instead of additional inclusive).

Share: Share of associate degree itemset is that the quantitative relation of the count of things purchased along to the entire count of things purchased altogether transactions.

Confidence: Confidence of rule X ® Y, denoted conf(X ® Y). The arrogance worth of a rule, X ⇨Y, with relevance a group of transactions T, is that the proportion the transactions that contains X that additionally contains Y. Confidence is outlined as:

conf(X ⇨Y) = supp(X ∪Y)/supp(X)

For example, the rule   ⇨ encompasses a confidence of zero.2/0.2 =1 within the information, which suggests that for 100 percent of the transactions containing butter and bread the rule is correct (100% of the days a client buys butter and bread, milk is bought as well). Note that supp(X ∪Y) suggests that the support of the union of the things in X and Y. this is often somewhat confusing since we have a tendency to usually assume in terms of possibilities of events and not sets of things. We will rewrite supp(X ∪Y) because the probability P ($E_X \cap E_Y$), wherever E_X and E_Y area unit the events that a dealing contains itemset X or Y, severally.

Transaction information: Transaction information stores dealing information. Transaction information may additionally be hold on in another type than associate degree m x n information.

High Utility Itemset Mining

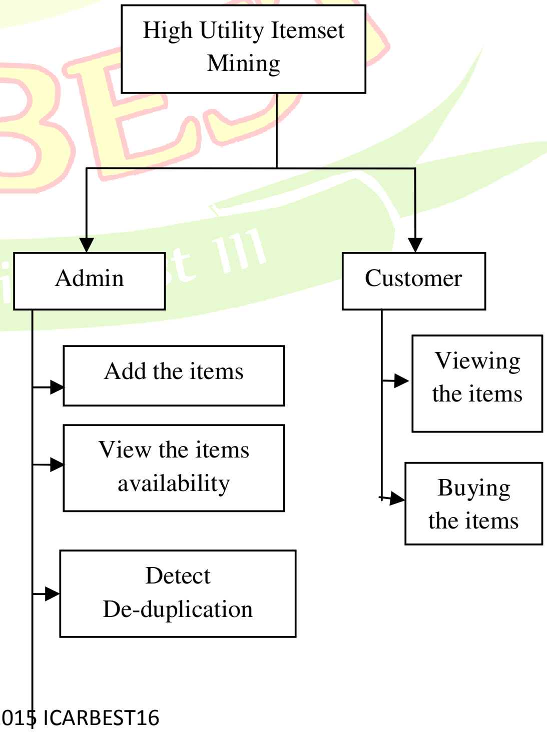
Let $I=\{i_1,i_2,....,i_M\}$ has a set of items, and $D=\{T_1,T_2,.....,T_N\}$has a set of transactions database. Every dealing T_R has belongs to dealing information D and variety of thingsets I to get things i as a result of all items area unit hold on in information D, wherever amount of itemsets is portrayed as QT(I). To spot every dealing distinctive symbol is employed. The inner utility is outlined because the items i will count within the transactions information D for every transactions T_R. The external utility is outlined because the utility values area unit accessory within the information D for each dealing. High utility itemset is outlined as set of things i no but user specified minimum threshold; aside from high utility itemset is named as low utility itemset. The length of the itemset is denoted as M variety of item in an exceedingly set. i.e. $1 \leq l \leq M$.

4. Proposed System

The traditional ARM approaches deal with the utility of the items by its presence in the transaction set. The frequency of itemset is not acceptable to reflect the actual utility of an itemset. For example, the sales manager may not be attentive in frequent itemsets that do not archive significant profit. Recently, one of the most challenging data mining tasks is the mining of high utility itemsets are more efficiently. Identification of the itemsets with high utilities is called as Utility Mining.

The measurement of utility can be in terms of cost, profit or other expressions of user preferences. For example, in a computer system may be leads more valuable than a telephone in terms of profit. Utility mining model was proposed in to characterize the utility of itemset. The utility is a measure of how useful or profitable an itemset X is. The utility of an itemset X, i.e., u(X), is the amount of the utilities of itemset X in all the transactions containing X. An itemset X is called as high utility itemset if and only if u(X) >= min_utility, where min_utility is a end users specified minimum utility threshold. The main objective of high utility itemset mining is to find all those itemsets having utilities are greater or equal to end users specified minimum utility threshold. Several data mining tasks are based on the evaluation of frequency queries to resolve how often a particular pattern occurs in a large data set. This project considers the problem of frequency query evaluation, when patterns are itemsets or conjunctions of properties, in dense datasets like, for instance in the context of census data analysis or log analysis. In these important but difficult purposes, there is a combinatorial explosion of the number of frequent itemsets and computing the frequency of all of them turns out to be intractable.

This project presents an efficient extraction of patterns that have good odds ratio exploiting convexity, we formulate a number of solid and complete algorithms to extract the most general and the most definitive of such patterns at each support level. The odd ratio pattern algorithm is able to mine these refined patterns at a speed comparable to that of mining frequent closed patterns.



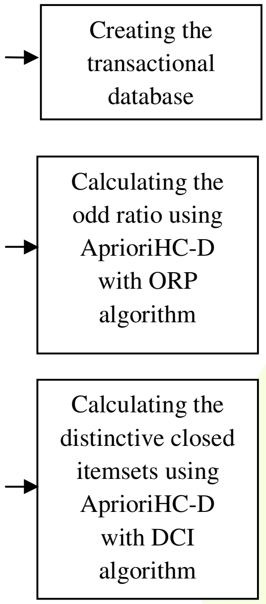

Creating the transactional database
Calculating the odd ratio using AprioriHC-D with ORP algorithm
Calculating the distinctive closed itemsets using AprioriHC-D with DCI algorithm

Fig 1 System Architecture

In the representation proposed in this paper, we are often interested to test whether a given principle has a given issue. If we cannot specify the nature of the factors convoluted, such tests are called as model-free studies. Fig 1 shows the system architecture of high utility mining. There are two modules presents in high utility mining. One is admin module and other one is customer module. The admin module consists of adding the items, viewing the items, detect de-duplication, creating transactional database for all available items, calculating the odd ratio pattern using AprioriHC-D with ORP algorithm and calculating the distinctive closed itmsets using AprioriHC-D with DCI algorithm. The customer module consists the viewing the items and buying the items.

4.1Odd Ratio pattern

In contrast with a "case-control" study first find the individuals by product status and then tests retrospectively for exposure to an aspect. In such a situation, the number of selling products (called the "principles") and non-selling products (called the "issues") is fixed, and thus the risk that an exposed customer will become a selling products case cannot be estimated directly. In Odd Ratio pattern, we proposed the ratio of the odds that a principle has been exposed to the odds that an issue has been exposed. This ratio is called the "odds ratio". Using the notations in Figure 1, the odds ratio for a factor C in a "case-control" study is

$$OR(C,SP) = \frac{\frac{\quad}{\quad}}{\quad}$$

Where C_{SP},ed is the number of individuals in SP who have disclosure to C and have developed the selling products, C_{SP},e− is the number of individuals who have disclosure and have not developed the selling products, C_{SP},−d is the number of individuals who have no disclosure and have developed the selling products, and C_{SP},−− is the number of individuals who have no disclosure and have not developed the selling products.

Under the null hypothesis, we expect the odds ratio equal to 1. As OR(C,SP) lies in the range [0,∞], it is often transformed to the range [−1,1] as (OR(C,SP)−1)/(OR(C,SP) + 1).

Discovering all frequent Odd Ratio pattern

This section defines an algorithm, called AprioriHC-D with ORP that generates all frequent itemsets. For clarity, this method omits the fact that it outputs their supports as well. In odds ratio patterns, we consider only dataset D whose transactions are denoted either as "positive" or as "negative". Given a dataset D, we write D^{pos} for those transactions in D that are assumed as positive, and D^{neg} for those assumed as negative. For our discussion on odds ratio, we take D^{pos} to be the selling products and D^{neg} to be the not-selling products.

```
Algorithm4.1(AprioriHC-D-ORP)
Input:DatasetD=$D^{pos}$U$D^{neg}$, threshold for supportms,and threshold for odds ratiok.
Output:A concise representation of equivalence classes ofpatterns having support atleastmsand odds ratio atleastk. The concise representation comprises, for eachequivalence class, its borders, its support in$D^{pos}$, its support in$D^{neg}$,and its odds ratio.
Method:
    1:ε:= the collection of all equivalence classes of$F(ms,D)$,concisely represented by their borders, and annotatedwith their support levels.
    2:foreach ‹L,{R}› inEdo
    3:x:=OR(R,D);
    4:y:=sup(R,$D^{pos}$);z:=sup(R,$D^{neg}$);
    5:ifx≥kthen
    6: output ‹ L,{R}›, x, y andz.
```

Since all patterns in $[P]_D$ must exist in exactly the same transactions in D, these patterns must have the same odds ratio value as P. Hence we can extend our notations and write sup(E,D) and ORP(E,D) respectively for the support and odds ratio value of an equivalent class E in D. The first approach is to first identify all the equivalence classes along with their borders in D, and filter them by their support in D^{pos} and by their odds ratio. The second approach is to first obtain a decomposition of the frequent itemsets in D^{pos}, then extract the border of the corresponding to the support level of in D^{pos}. In each element, R_j in the right bound of the closed pattern of an equivalence class. For each R_j, we check also D^{pos} to see if this R_j have good odds ratio. For each R_j, we check also D^{neg} to see if this R_j have good odds ratio. If it does not have good odds ratio, then its entire equivalence class can be discarded. If it has good odds ratio, then we need to consider the corresponding left bound L_j. We need to move L_j right to a more specialized L'_j so that the requirements on odds ratio are satisfied. The algorithm is given above as Algorithm 4.1.

4.2AprioriHC-D-DCI

The Minimum Description Length (MDL) principle finding Concise (or condensed) representations of frequent patterns, by giving the shortest description of the whole set of frequent patterns. In this work, we suggest a new exact concise representation of frequent itemsets. This representation is established on an exploration of the disjunctive search space. The disjunctive itemsets bring information about the complementary occurrence of items in a dataset. A novel closure operator is then arranged to suit the characteristics of the explored search space. The proposed operator aims at mapping many disjunctive itemsets to a unique one, they are termed as disjunctive closed itemset. Hence, it permits to drastically reduce the number of managed itemsets within the targeted re-presentation. Interestingly, the proposed representation offers direct connection to the disjunctive and negative supports of frequent itemsets while ensuring the derivation of their exact conjunctive supports.

To the best of our knowledge, the exact concise representation based on frequent essential itemsets is the unique representation contributes this interesting feature through its exploration of the disjunctive search space. In this space, itemsets are identified by their respective disjunctive supports. Thus, an itemset verifies an element of a dataset (or transaction) if one of its items exists to this transaction. With respect to set inclusion, an essential itemset is the minimal set of items among those itemsets representing a common set of transactions. There are three different supports are as Conjunctive support, Disjunctive support and Negative support.

Algorithm 4.2

```
AprioriHC-D-DCI($I_i$,h)
Input: $I_i$={$i_1$,$i_2$,…,$i_n$} be a set of itemsets, h be the closure operator.
Output: Disjunctive support follows that Supp(I) is the number of transactions containing at least one item of I. Negative support follows that Supp(I) is the number of transactions that do not contain any item of I. An itemset I is said to be frequent if Supp(I) is greater than or equal to a minimum support threshold, denoted minsupp. This representation can solves straightforwardly be done in a levelwise manner that regenerates 1-frequent itemsets, 2-frequent itemsets, and so forth.
h:= 1; DCI:= {}; C1:= {{h} | h Є I};
for all I in $C_1$ do I.l:= 0; I.u:= |$I_i$|;
while$C_i$ not empty do
Count the supports of all candidates in $C_i$ in one pass over D;
$F_i$:= {I Є $C_i$ | Supp($I_i$,h) ≥ s};
DCI:= DCI ∪ $F_i$;
Gen:= {};
for all I Є $F_i$ do
ifSupp(I) ≠ I:l and Supp(I) ≠ I:u then
Gen:= Gen ∪ {I};
$PreC_{h+1}$:= AprioriGenerate(Gen);
$C_{h+1}$:= {};
for all J Є $PreC_{h+1}$ do
Compute bounds [l, u] on support of J;
if l ≠ u then J, l:= l; J.u:= u;$C_{h+1}$ := $C_{h+1}$∪ {J};
h:= h + 1
end while
return DCI
```

Let us consider I_i={i_1,i_2,…,i_n} be a set of itemsets. Conjunctive support follows that Supp(I_i) is the number of transactions containing all items of I. Disjunctive support follows that Supp(I_i) is the number of transactions containing at least one item of I. Negative support follows that Supp(Ii) is the number of transactions that do not contain any item of I. An itemset I is said to be frequent if Supp(I_i) is greater than or equal to a minimum support threshold, denoted minsupp. The essential idea of this new concise representation is to apply a closure operator *h* on frequent essential itemsets to access a more compact concise representation while preserving their interesting properties.

The disjunctive itemsets will be converted into subsets, and each subset simply characterized by a unique element is the disjunctive closed itemset. We examine that *h* be the closure operator that structurally characterizes the disjunctive closure of any itemset I_i. This allows to straightforwardly computing disjunctive closed itemsets freely either using either of breadth-first or a depth-first traversal of the search space. With the help of closure operator *h*, the disjunctive search space is separated into the so-called disjunctive equivalence classes. In each class, the associated disjunctive closed itemset is the unique maximal element, while the corresponding essential itemsets are the minimal ones. The elements of each equivalence class interest the same set of transactions and thus have the same disjunctive support and disjunctive closure. The examination of the disjunctive search space using the operator *h* will be at the origin of the advanced concise representation for the set of frequent itemsets as explained in the remainder. The algorithm shown as above in figure 4.2

5. SCREENSHOTS

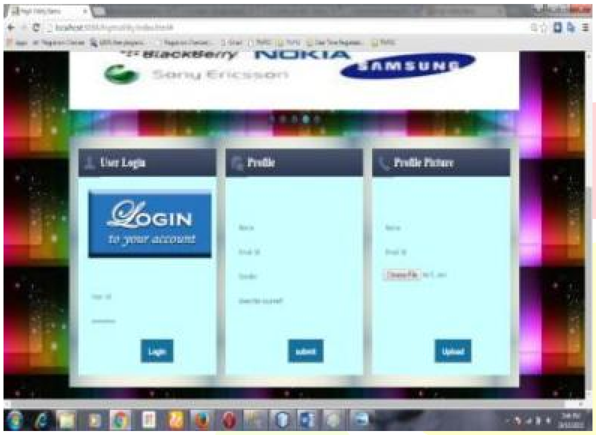


Fig 2.Home Page

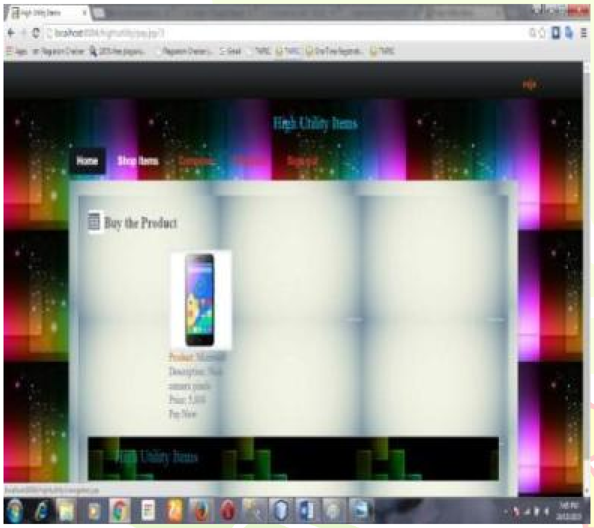


Fig 3.Customer Page

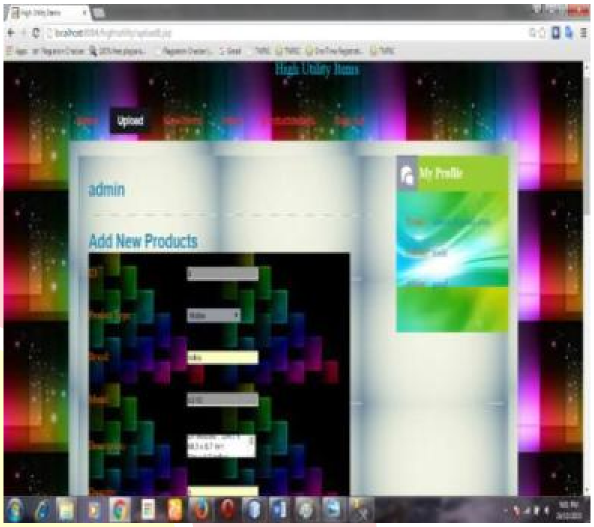
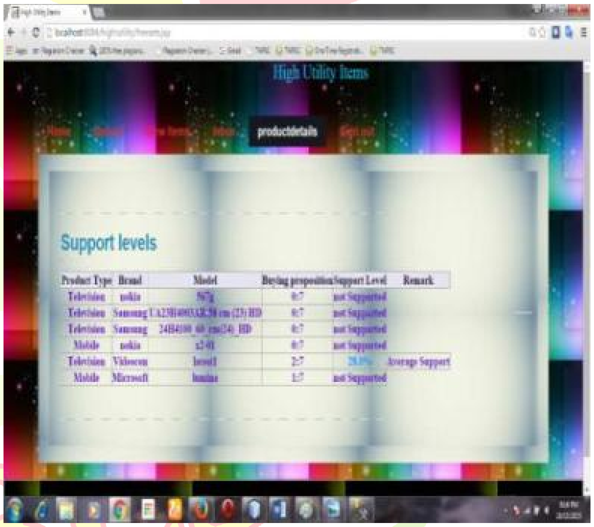


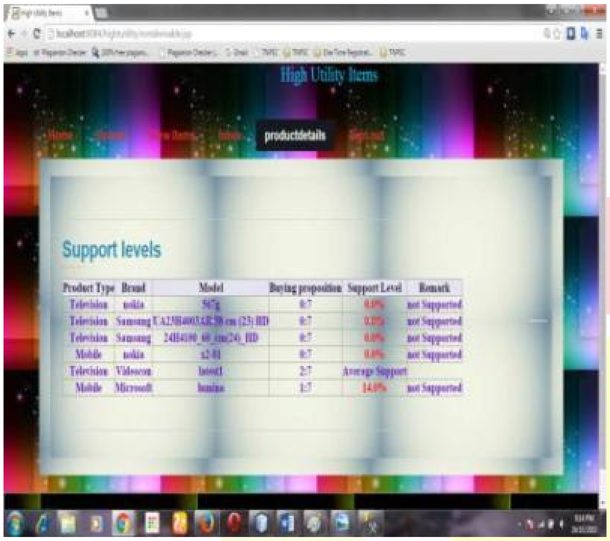
Fig 4.Admin Page



Support levels

Product Type	Brand	Model	Buying proposition	Support Level	Remark
Television	nokia	567g	0:7	not Supported	
Television	Samsung	UA23H4003AR 58 cm (23) HD	0:7	not Supported	
Television	Samsung	24H4100_60_cm(24)_HD	0:7	not Supported	
Mobile	nokia	x2-01	0:7	not Supported	
Television	Videocon	lava1	2:7	28.5%	Average Support
Mobile	Microsoft	lumia	1:7	not Supported	

Fig 5.AprioriHC-D-ORP



Support levels

Product Type	Brand	Model	Buying proposition	Support Level	Remark
Television	nokia	567g	0:7	0.0%	not Supported
Television	Samsung	UA23H4003AR 58 cm (23) HD	0:7	0.0%	not Supported
Television	Samsung	24H4100_60_cm(24)_HD	0:7	0.0%	not Supported
Mobile	nokia	x2-01	0:7	0.0%	not Supported
Television	Videocon	latest1	2:7	Average Support	
Mobile	Microsoft	lumina	1:7	14.0%	not Supported

Fig 6.AprioriHC-D-DCI

6. Conclusion

In this project for mining high utility itemsets two algorithms are presented namely AprioriHC-MINEX (AprioriHC-D algorithm with ORP) and AprioriHC-D-DCI (AprioriHC-D algorithm with Distinctive Closed Itemsets). We introduce that it outputs theirsupports is well. This experiments also show that the error made when approximatingthe support of frequent itemsets using the support of odd ratio pattern remains verylow in practice. Finally, we considered the effect of this approximation on the supportand confidence of association rules.These two algorithms are uses AprioriHC-D which perform a breadth-first search for calculating high utility itemset mining. The results shows that these two algorithms efficiently calculating the high utility itemsets.

References

[1]Alva Erwin, Raj P. Gopalan, and N.R. Achuthan (2011), 'Efficient Mining of High Utility Itemsets from Large Datasets', Springer-Verlag Berlin Heidelberg, Advances in Knowledge Discovery and Data Mining, Volume 5012 of the series Lecture Notes in Computer Science, pp. 554-561.

[2]Bai-En Shie, Hui-Fang Hsiao, Vincent S. Tseng, and Philip S. Yu (2008), 'Mining High Utility Mobile Sequential Patterns in Mobile Commerce Environments', Database Systems for Advanced Applications, Volume 6587 of the series Lecture Notes in Computer Science. pp. 224-238.

[3]Cheng Wei Wu, Philippe Fournier-Viger, Philip S. Yu, Vincent S. Tseng (2011), 'Efficient Mining of a Concise and Lossless Representation of High Utility Itemsets', 11th IEEE International Conference on Data Mining, pp. 824-833.

[4]Ching-Huang Yun and Ming-Syan Chen (2007), 'Mining Mobile Sequential Patterns in a Mobile Commerce Environment', IEEE Transactions on Systems, Man, and Cybernetics Society, Vol. 37, pp. 278 – 295.

[5]Claudio Lucchese, Salvatore Orlando, and RaffaelePerego (2006), 'Fast and Memory Efficient Mining of Frequent Closed Itemsets', IEEE Transactions on Knowledge and Data Engineering, Vol. 18, pp. 21 – 36.

[6]Erwin, R. P. Gopalan, and N. R. Achuthan (2008), 'Efficient mining of high utility itemsets from large datasets,' in Proc. Int. Conf. Pacific-Asia Conf. Knowl. Discovery Data Mining, pp. 554–561.

[7]Frequent itemset mining implementations repository http://fimi.ua.ac.be/data/

[8]Guo-Cheng, LanTzung-Pei Hong, Vincent S. Tseng (2011), 'Projection-Based Utility Mining with an Efficient Indexing Mechanism', International Conferences on Technologies and Applications of Artificial Intelligence (TAAI), pp. 137 – 141.

[9]Jean-Fran_coisBoulicaut, ArturBykowski, and Christophe Rigotti (2010), 'Approximation of Frequency Queries by Means of Free-Sets', Springer-Verlag Berlin

Heidelberg, Principles of Data Mining and Knowledge Discovery, Volume 1910 of the series Lecture Notes in Computer Science, pp. 75-85.

[10]J. Han, J. Pei, and Y.Yin (2000), 'Mining frequent patterns without candidate generation,' in Proc. ACM SIGMOD Int. Conf. Manage. Data, pp. 1–12.

[11]Kun-Ta Chuang, Jiun-Long Huang† and Ming-Syan Chen (2011), 'Mining Top-k Frequent Patterns in the Presence of the Memory Constraint', IEEE Transactions on System, the VLDB Journal, Volume 17, pp. 1321-1344.

[12]R. Agrawal and R. Srikant (1994), 'Fast algorithms for mining association rules', in Proc. 20th Int. Conf. Very Large Data Bases, pp. 487–499.

[13]R. Chan, Q. Yang, and Y. Shen (2003), 'Mining high utility itemsets', in Proc. IEEE Int. Conf. Data Min., pp. 19–26.

[14]T. Hamrouni, S. Ben Yahia, E. MephuNguifo (2010) 'Sweeping the disjunctive search space towards mining new exact three concise representations of frequent itemsets', Journal Data & Knowledge Engineering, Volume 68, pp. 1091-1111.

[15]Ying Liu, Wei-keng Liao, AlokChoudhary (2009), 'A Fast High Utility Itemsets Mining Algorithm', ACM transaction System, UBDM '05 Proceedings of the 1st international workshop on Utility-based data mining, pp. 90-99.