Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 3, March 2014, pg.889 – 893

RESEARCH ARTICLE

FUFM-High Utility Itemsets in Transactional Database

S.Priya[*1], E.Thenmozhi[*2], Mrs. D.Shiny Irene[#3]

[*1,*2]U.G Student, B.E CSE, Alpha College of Engg, Chennai, T.N, India
[#3]Assistant Professor, Dept. of CSE, Alpha College of Engg, Chennai, T.N, India
priya45sweety@gmail.com

Abstract – The practical usefulness of the frequent item set mining is limited by the significance of the discovered itemsets. There are two principal limitations. A huge number of frequent item sets that are not interesting to the user are often generate when the minimum support is low.Proposing two algorithms, namely utility pattern growth (UP-Growth) and UP-Growth+, for mining high utility itemsets with a set of effective strategies for pruning candidate itemsets.

Keywords – Candidate pruning, frequent itemset, high utility itemset, utility mining, data mining

I. INTRODUCTION

Data mining and knowledge discovery from data bases has received much attention in recent years. Data mining, the extraction of hidden predictive information from large databases, is a powerful new technology with great potential to help companies focus on the most important information in their data warehouses.

The Knowledge Discovery in Databases (KDD) is the non-trivial process of identifying valid, previously unknown and potentially useful patterns in data. These patterns are used to make predictions or classifications about new data, explain existing data, summarize the contents of a large database to support decision making and provide graphical data visualization to aid humans in discovering deeper patterns.

The KDD process comprises of a few steps leading from raw data to some form of new knowledge. The volume of data contained in a database often exceeds the ability to analyze it efficiently, resulting in a gap between the collection of data and its understanding.

In knowledge discovery, techniques are constantly being developed and improved for discovering various types of patterns in databases. While these techniques were shown to be useful in numerous applications, new problems have also emerged.

II. RELATED WORKS

The problem of [1] Problem is that each entry may be larger than the corresponding transaction.

Solution is using Apriori hybrid algorithms. The problem of[2] is mining sequential patterns over a large database of customer transaction.

Solution is using apriorisome and aprioriall. The problem of[9] is discovering frequent patterns in databases with multiple time series and the solution is proposing an incremental technique for discovering the complete set of frequent patterns, i.e., discovering the frequent patterns over the entire time series in contrast to a sliding window over aportion of the time series. The problem of[10] is mining high utility itemsets presents a greater challenge than frequent itemset mining, since high utility itemsets lack the anti-monotone property of frequent itemsets and the solution is presenting the *CTU- PROL* algorithm *to* mine the complete set of high utility itemsets from both sparse and relatively dense datasets with short or longer high utility patterns.

III. DESIGN AND IMPLEMENTATION

A. Customer Registration:

The customers who want to buy products must register with his/her personal information to the application. The user information are stored in the database and to maintain the user profile. To gain access to the application the user must provide authentication details to the application. The information entered is compared with the information (of the particular customer) stored in the database; if they match then he is give a right to access the next page otherwise not.

B. Product Purchase/Transaction

The objective is to divide the whole transaction database into parts that can be mined independently. In this module the customer sees the products available and gives his order by selecting his/her desired product. The order is stored in the database only after he presses the submit button and next process is carried out. We can purchase the enterprise products.

C. Utility and Frequent Utility Mining

Utility based and has a pruning strategy of its own. Its goal is High utility itemset mining. The UMining algorithm follows the basic framework of the Apriori algorithm, but there are significant differences in three sub functions

(Prune, CalculateAndStore, and Generate functions). Utility of all itemsets and their combinations are obtained and High Utility Itemsets are mined. From this threshold we need to identify the Minimum utilized product.

Frequent Utility Mining (FUM) algorithm generates high utility itemsets using Combination Generator. It is simpler and executes faster than UMining algorithm.

CombinationGenerator(T) - Generate all possible combinations of itemset $\in$ T

D. Frequent Utility Frequent Mining(FUFM)

Frequent Utility-Frequent Mining(FUFM) which finds all utility-frequent itemsets within the given utility and support constraints threshold. Utility-frequent itemsets are a special form of high utility itemsets using Selective Item Replication.

a.HUHF: High utility and high frequency itemsets by incorporating support into FUM algorithm.First phase of this algorithm is to generate high utility itemsets H.

b.HULF: high utility and low frequent itemset by support both FUM and FUFM algorithms.

- The first phase is to generate high utility itemsets using FUM algorithm.
- The second phase high utility high frequent itemsets are generated using FUFM(HU).

c.LUHF: To generate Low utility and high frequent itemsets. It follows the basic frame work of FUFM algorithm.

d.**LULF: Low utility and low frequent**

- First phase using exhaustive search low utility itemsets are determined.
- Second phase, using set difference function low utility low frequent itemsets are generated from LU and LUHF.

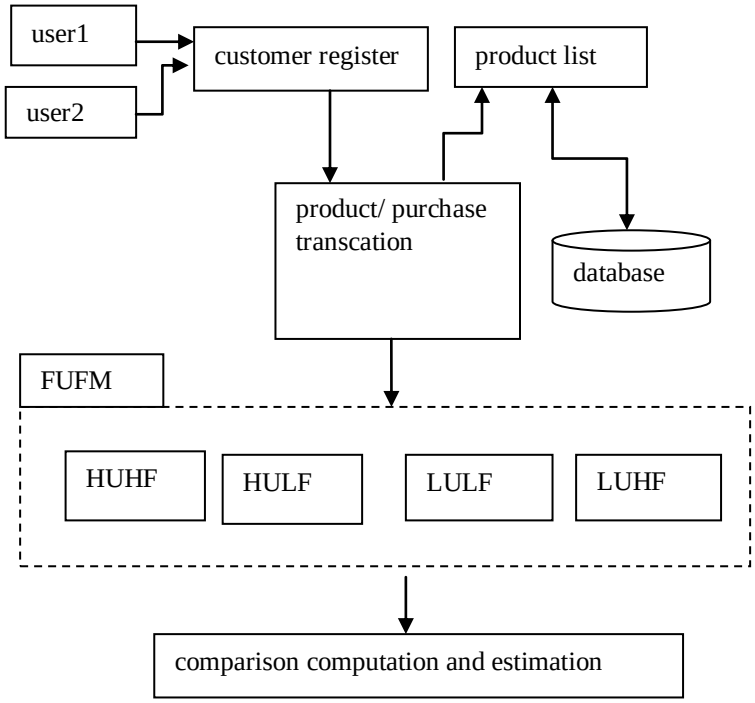
$T_{S,\mu} = \{T|S \subseteq T \wedge u(S, T) \geq \mu \wedge T \in DB \}.$

$$\text{support}(s,\mu)=\frac{\left|T_{s,\mu}\right|}{\left|DB\right|}$$

E. Computation & Estimation

To minimize collective work. If the computational work estimate for a projection over a set of items X is in the form of a summation of individual load estimates for items. Our proposed GPVS model will minimize collective work instead of minimizing data replication. It will also balance computational load.

fig1:sysytem architecture



IV. CONCLUSION & FUTURE WORK

The proposed FUM algorithm executes faster than existing Utility mining algorithm, when more itemsets are identified as high utility itemsets. A new algorithm FUFM, is designed to generate HUHF itemsets which makes use of both utility and frequency, and compared with FUFM. The experimental evaluation on artificial datasets proved that FUFM algorithm is better than the proposed FUM-F algorithm in terms of execution time and pruning strategy. Thus FUM and FUFM algorithms are used to generate the remaining itemsets HULF, LUHF and LULF and the algorithms are evaluated by applying them to databases

REFERENCES

[1] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules," Proc. 20th Int'l Conf. Very Large Data Bases(VLDB), pp. 487-499, 1994.

[2] R. Agrawal and R. Srikant, "Mining Sequential Patterns," Proc.11th Int'l Conf. Data Eng., pp. 3-14, Mar. 1995.

[3] C.F. Ahmed, S.K. Tanbeer, B.-S. Jeong, and Y.-K. Lee, "Efficient Tree Structures for High Utility Pattern Mining in Incremental Databases," IEEE Trans. Knowledge and Data Eng., vol. 21, no. 12,
pp. 1708-1721, Dec. 2009.

[4] C.H. Cai, A.W.C. Fu, C.H. Cheng, and W.W. Kwong, "Mining Association Rules with Weighted Items," Proc. Int'l Database Eng. and Applications Symp. (IDEAS '98), pp. 68-77, 1998.

[5] R. Chan, Q. Yang, and Y. Shen, "Mining High Utility Itemsets," Proc. IEEE Third Int'l Conf. Data Mining, pp. 19-26, Nov. 2003.

[6] J.H. Chang, "Mining Weighted Sequential Patterns in a Sequence Database with a Time-Interval Weight," Knowledge-Based Systems, vol. 24, no. 1, pp. 1-9, 2011.

[7] M.-S. Chen, J.-S. Park, and P.S. Yu, "Efficient Data Mining for Path Traversal Patterns," IEEE Trans. Knowledge and Data Eng., vol. 10,no. 2, pp. 209-221, Mar. 1998.

[8] C. Creighton and S. Hanash, "Mining Gene Expression Databases for Association Rules," Bioinformatics, vol. 19, no. 1, pp. 79-86,2003.

[9] M.Y. Eltabakh, M. Ouzzani, M.A. Khalil, W.G. Aref, and A.K.Elmagarmid, "Incremental Mining for Frequent Patterns inEvolving Time Series Databases," Technical Report CSD TR#08-02, Purdue Univ., 2008.

[10] A. Erwin, R.P. Gopalan, and N.R. Achuthan, "Efficient Mining of High Utility Itemsets from Large Data Sets," Proc. 12th Pacific-AsiaConf. Advances in Knowledge Discovery and Data Mining (PAKDD),pp. 554-561, 2008.

