Proposed Design for Data Retrieval using Efficient Algorithm

Kajal Kokane, S.M. Chaware

Abstract**: *Data mining is used for finding patterns from large amount of data which is in raw format. These patterns are then analyzed to gain useful information from them. There are many branches of data mining, one of the most interesting branch is frequent item-set mining (FIM). FIM deals with finding items that are frequently brought together by customers. Like for example, if a customer purchases a mobile phone, he also tends to purchase mobile cover, ear phones etc along with it. But such kinds of patterns are not always useful to all stake-holders. Such patterns do not emphasize on the profit obtained of sale i.e. the utility obtained from product. In order to overcome this problem, the concept of high utility item-set mining (HUIM) came into existence. HUIM is used to find the utility or profit obtained from the items in transaction data. There are various algorithms for HUIM, TKU (Top K Utility) and TKO (Top K in one phase) are two well known algorithms of HUIM. The detailed study and practical analysis of these two algorithms show that there are certain drawbacks assigned with them. TKO algorithm gets executed in very less amount of time but it gives incorrect output. Whereas TKU algorithm gives accurate results when applied on database, but its execution time is very high. Hence in order to enhance the performance of these two HUIM algorithms a hybrid algorithm i.e. TKO with TKU algorithm is proposed in this paper. The two algorithms when combined give accurate result and also get executed in considerable less amount of time***

Keywords: High Utility Item-set, (mining Top-K utility in one phase) TKO, (mining Top-K utility item-sets) TKU, Data mining.

I. INTRODUCTION

A. Background

High utility item-set mining is the next step to frequent item-set mining. FIM deals with finding frequent and continuous patterns in the database. Frequent item-set mining has proved to be very useful in many industrial applications like the market basket analysis. There are various algorithms for finding the frequently bough items like ECLAT, FP-Growth, Apriori etc. These algorithms use the parameter called as min-sup for all the calculations. But FIM is not always useful in all aspects of research. Consider if some customer buys a product of very less price, like a packet of pen, and also the customer buys some other expensive item like a wrist watch. The FIM algorithm considers items, pen and wrist watch as having same weightage or same utility. So such kind of transaction gets un-noticed and the profit constraint gets completely omitted. Also if a customer buys one single item like one single pen and also the customer buys 15 pens together, yet these two transactions are considered as same.

The quantity of product is not considered in FIM. Thus, FIM may find patterns that are not of any use to the stake-holders. Uninteresting patterns from business and industrial perspective may be detected. Also certain patterns which are of high benefit to the organization can be missed. To get such useful patterns from the transaction database which can find out highly profitable transaction the concept of HUIM is considered. HUIM are capable of finding such items which can be of high use and produce more and more profit for the organization or industry.

B. Motivation

Frequent item-set mining algorithms like Apriori are quite famous, but they have many limitations. One of the limitations is, count of number of items bought is not considered [14]. Other major drawback is that all items are viewed as having the same importance or utility of weight [10].

Hence to overcome these limitations, High utility item-set mining concept can be used. The two algorithms of High utility item-set mining are TKO and TKU. But these algorithms are not efficient. They have their own drawbacks like incorrect output and extremely high time for execution. So in order to improve the efficiency a hybrid algorithm has been proposed for mining high utility item-sets accurately.

C. Objectives

1) To implement the hybrid algorithm TKO WITH TKU for improving accuracy of high utility mining.

2) Reduce time taken by algorithm to get executed by providing output of one algorithm as input to other algorithm.

3) Improve the efficiency of high utility mining.

II. REVIEW OF LITERATURE

Frequent item-set mining is the study of finding out which items are purchased hand in hand with each other. FIM is a a very wide branch of data mining and has many different algorithms for finding the set of frequent item-sets which belong together in the transactional database. Various algorithms like FP-Growth, ECLAT i.e. Equivalence Class Transformation, Apriori etc are used for mining of frequently together occurring items. These three algorithms show different performance when compared with respect to time and space [1]. Apriori algorithm requires highest time for execution but at the same time it requires least storage space. FP Growth algorithm requires least time for execution and moderate space complexity, Whereas Eclat algorithm requires moderate time of execution but highest space complexity if all three algorithms are compared.

Revised Manuscript Received on October 30, 2019.

* Correspondence Author

Kajal Kokane*, B.E Department of Computers, Padmabhooshan Vasantraodada Patil Institute of Technology, Mumbai, (Maharashtra), India.

S.M. Chaware, Ph.D., NMIMS University, Pherozeshah Mehta Rd, Vile Parle West, Mumbai, (Maharashtra), India.

Association rules are extracted from the transactional database once apriori algorithm is applied to the database. There are many applications and recommendation systems which use association rules for finding frequent patterns. Book recommendation system is one of the applications where association rule mining is applied in order to recommend books to customers [2]. Before rule mining is performed, data pre-processing steps are performed on the data. Which include, finding empty or blank data which is also called as missing values and filling those values with some default values. Also data pre-processing includes finding and removing data outliers. And after pre-processing step, the association rule mining algorithm is applied to get the final output.

Association rule mining can be applicable in other similar kind of recommendation systems. It has been applied to library book recommendation system which is based on user profiles [3]. Recall, confidence, support such calculations are involved in this system.

High utility item-set mining is the next step to frequent item-set mining concept and it overcomes the various drawbacks of FIM. HUIM gives as output utility based items according to user specified utility. No min-sup value is required by HUIM algorithms. TKO and TKU are two well known algorithms used for HUIM [4]. These algorithms include finding the potentially K high utility item-sets as its basic step. The execution time required by TKU algorithm is very high as compared to TKO algorithm. But TKU algorithm is more efficient as it gives better results as compared to TKO algorithm.

Mining of top-k HUI also has algorithms other than TKO and TKU like Top k utility list miner (TKUL-Miner) [5]. Utility list structure is used for mining user specified utility depending upon the k value given by user. Information is stored at the node of search tree and then mining operation is performed using TKUL-Miner algorithm.

Isolated item discarding strategy i.e. IIDS is one of the strategies used for discovering the HUI's [6]. The isolated items are those items which are of very less importance with business perspective. The items which hold very less frequency count and with low utility are discarded i.e. they are not considered for further mining process. Thus discarding these isolated items finally leads as output the items having maximum utility.

Up-tree gets generated dynamically and it is used to store the data in a tree format [8]. As soon as the data gets updated by the user it is stored in up growth table. Due to this there is no need to touch the original database again and again. This avoids accessing the original database multiple times. UP-Tree analyzes the database only two times. Once to have candidate elements and then second time to manage them in a structured and efficient manner. Using up-tree, Up-growth algorithm shows poor performance with respect to time. It requires very high time for execution using the up tree. Hence, modified and better algorithms are proposed which effectively reduce the time required for execution.

There are so many different algorithms available for mining frequent item-sets or for mining high utility item-sets, still the fact that no algorithm has all the required aspects comes into existence [11]. Some strategies used for mining lead to accuracy in results but fail when time required for execution is considered. This condition can be worse if the amount of data scanned by these algorithms increases. For more amounts of data it will take far more time for execution. In this paper, two calculations are proposed, to be specific, utility example development (UP-Growth) and UP-Growth+, for mining high utility thing sets with an arrangement of successful methodologies for pruning hopeful thing sets.

High utility thing sets data is kept up in a tree based information structure named utility example tree (UP-Tree) to such an extent that applicant thing sets can be created effectively with just two sweeps of database. The execution of UP-Growth and UP-Growth+ is differentiated on various sorts of both honest to goodness and designed instructive accumulations. Theoretical calculations show that the proposed algorithm, especially UP-Growth+, decrease the candidates count enough and also beat diverse counts efficiently to the extent runtime, especially when databases is huge.

Below listed re few limitations observed from the overall literature survey:

1) For large databases existing algorithms show poor performance.

2) Execution time required in case of huge transactions is very high for existing algorithms.

3) Results in case of certain algorithms may be inaccurate i.e. garbage values.

4) In case of frequent item-set mining algorithms like FP-Growth, Apriori etc. purchase quantities are not at all taken into account.

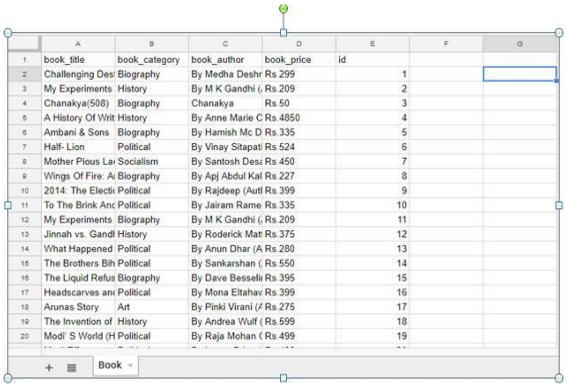
5) Frequent pattern mining finds many frequent patterns that are not interesting from industrial point of view [13].

III. PROPOSED METHODOLOGY

In the proposed framework, the problems mentioned above are considered and a book recommendation system is implemented using the already existing two algorithms TKO and TKU and also using the proposed hybrid TKO with TKU algorithm. Since the two existing algorithms are not efficient enough due to incorrect results and more than expected time of execution, the hybrid algorithm is designed by combining the two algorithms i.e. the result of TKO Top K in one phase is given at the entrance of TKU Top K utility algorithm. This results in accurate result in considerably less execution time. Thus the proposed system combines the two HUIM algorithms to enhance the performance of mining. Parameters like k value i.e. number of books the user needs to extract from the transactions is to be given by user, along with the category of book i.e. historic, romantic, fiction etc.

In order to implement it, a books dataset consisting of total 2590 records is used. The headers of this dataset are as follows:

1) Book Title
2) Book Category
3) Boo Author
4) Book Price
5) Id



	A	B	C	D	E	F	G
1	book_title	book_category	book_author	book_price	id		
2	Challenging Des	Biography	By Medha Deshr	Rs.299	1		
3	My Experiments	History	By M K Gandhi (.	Rs.209	2		
4	Chanakya(508)	Biography	Chanakya	Rs.50	3		
5	A History Of Writ	History	By Anne Marie C	Rs.4850	4		
6	Ambani & Sons	Biography	By Hamish Mc D	Rs.335	5		
7	Half- Lion	Political	By Vinay Sitapati	Rs.524	6		
8	Mother Pious La	Socialism	By Santosh Des	Rs.450	7		
9	Wings Of Fire: A	Biography	By Apj Abdul Kal	Rs.227	8		
10	2014: The Electi	Political	By Rajdeep (Autl	Rs.399	9		
11	To The Brink Anc	Political	By Jairam Rame	Rs.335	10		
12	My Experiments	Biography	By M K Gandhi (.	Rs.209	11		
13	Jinnah vs. Gandl	History	By Roderick Mat	Rs.375	12		
14	What Happened	Political	By Anun Dhar (A	Rs.280	13		
15	The Brothers Bih	Political	By Sankarshan (	Rs.550	14		
16	The Liquid Refus	Biography	By Dave Besseli	Rs.395	15		
17	Headscarves an	Political	By Mona Eltahav	Rs.399	16		
18	Arunas Story	Art	By Pinki Virani (A	Rs.275	17		
19	The Invention of	History	By Andrea Wulf (	Rs.599	18		
20	Modi' S World (H	Political	By Raja Mohan (	Rs.499	19		

Fig. 1. Book Data Set Screenshot

Below listed are the advantages of Proposed System:

1. The issue of setting min-sup value is completely omitted with the use of TKO and TKU algorithms.
2. The proposed algorithm has less search space so it needs less memory.
3. It scans the database only one single time.
4. It is easy to implement.
5. Its performance is good in dense databases.

A. Architecture:

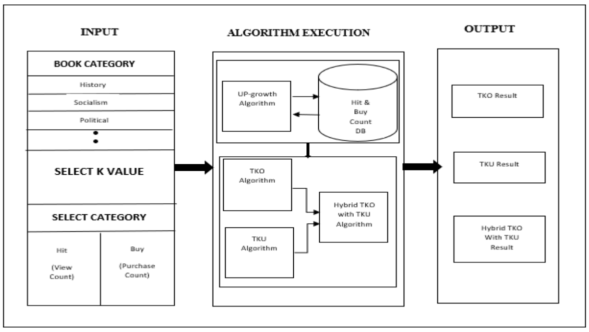


Fig. 2. Proposed System Architecture

Below is the explanation of above system architecture:

The above given system architecture illustrates that the end-user can select any category of book from the given list of categories, i.e. Historic, Romantic, Scientific, Fiction etc. The user also has to provide K-value. Here K-value refers to number of books the user needs to retrieve as having maximum hit count or maximum buy count depending upon the select category. Select category here either will be buy count i.e. number of times that particular book has been purchased, and hit count i.e. number of times, that book has been viewed.

Once the user provides all the specified inputs, the algorithms will be applied in order to get the result. Up-Growth algorithm is used to store Buy count and hit count values in a table. Initially the table will be empty, but as soon as some user views any product, the hit count will increment. If user will purchase any product, then value of buy count in UP-Growth table will get incremented by one. In this way, the values in UP-Growth table will always keep on incrementing. The value of hit count will always be less or equal to buy count.

The TKO with TKU algorithm will be then applied on the data. Output of TKO is given as the input of TKU and finally the result will be provided to user. If user selects Book category as Historic, K-value as 2 and select category as Buy, then as a result the user will get 2 historic books which have been purchased by other users maximum number of times.

Module 1 :Administrator (Admin)

The administrator is responsible to maintain database of the transactions made by users. The administrator also can add the product or items, and update the new product and also view the stock details.

Module 2 :User (Customer)

Customer can view or purchase the number of items. The history of all purchased items is stored in the transaction database.

Module 3 :User (Construction of Up Tree)

In Up Tree Dynamic Table is generated. Mainly the Up growth is considerable to get the PHUI item-set.

Module 4 :User (TKO and TKU Algorithms)

In combination of TKO and TKU algorithms first the TKO (Top k in one phase) algorithms is called and then output of TKO is given as the input of TKU (Top k in utility phases) and then the actual result is generated.

B. Algorithms:

TKO and TKU algorithm:

TKO stand for top-k in one phase and TKU stands for top k utility. These two algorithms are types of HUIM. HUIM finds out the patterns and those item-sets which produce high profitable transactions. HUIM has a vast domain of applications like it can used in any store, or online products systems to find items generating maximum profit or income. The database used for mining of HUI's should consider the item and number of time that item is purchased along with its utility.

Below given is the input database of TKO and TKU algorithms. It consists of 5 transactions, T1, T2, T3, T4, T5 with total of 6 different items i1, i2, i3, i4, i5, i6.

TABLE-I. Input to algorithm.

Transaction id	Items	Utility	Item utilities
T1	i1, i2,i5,i6	12	6,2,3,1
T2	i1,i5	9	6,3
T3	i3,i4,i2	13	7,4,2
T4	i4,i6,i5	8	4,1,3
T5	i3,i5	10	7,3

Table 1 describes the input database:

- Column 1: Transaction id.
- Column 2: A set of all items purchased by customer.
- Column 3: Utility obtained from corresponding transaction.
- Column 4: Last column describes item utility for each transaction i.e. profit generated by this item for the transaction.

The working of TKO and TKU algorithm is as follows:

The above given database i.e. table 1 is taken as input by TKO and TKU algorithms. As described above the values of utility column are the total of sum of values in last column.

Consider transaction T1, the item utility obtained in T1 is the addition of item utilities of T1 i.e. i1+i2+i5+i6 = 6 + 2 + 3 + 1= 12. In this way the item utilities of each transaction are calculated.

Once the input table is completely ready, it is used by the algorithms for further calculation.

Combination of different item-sets is generated and then, depending on the utility value of each item from input database the Final utility for that combination of transactions is calculated.

Consider the transaction {i1,i5}. This pair appears twice in the database i.e. in T1 and T2. So the total ulitity of that pair is calculated by adding the sole utility of each item i.e. 6 + 3 + 6 + 3 = 18. In this way transaction utilities of all different combinations of items are calculated and then the output table is generated accordingly. Finally the transaction utility with maximum value is considered as the HUI.

It is possible that there are more than one HUI's in a transactional data. The number of HUI's fetched depends upon the k value provided by the user.

TABLE- II. Output of algorithm

Item-sets	Utility
{i2,i4}	6
{i2,i5}	5
{i1,i3,i5}	0
{i2,i3,i4}	13
{i2,i3,i5}	0
{i1,i5}	18
{i1,i2,i5,i6}	12

If the database consists of data from grocery shop, the result can be depicted as all the k groups of items bought together that generated the maximum profit.

The only difference in TKO and TKU algorithms is the way of their execution. Both algorithms execute differently. The reading style of each variable in TKU is continuous whereas this is not the case for TKO algorithm.

Hybrid TKO with TKU algorithm:

The hybrid algorithm is proposed to overcome the limitations of TKO and TKU algorithms. The output of TKO algorithm is given as input to TKU algorithm. Hence it becomes possible to balance the algorithm and produce efficient result. In book recommendation system, the up growth table is maintained. It is a dynamic table. As soon as the user views or buys the book, the hit count or buy count increments by one. Once the up growth table is constructed, hybrid algorithm scans data one by one from that table. The algorithm reads the 1st row and writes it in other table. Simultaneously the records are sorted with highest buy or hit count at the top level of the table. The record with greater value is swapped with record having less value. Thus the output table gets constructed in decreasing order of buy or hit count. For getting accurate results without any incorrect values, the hybrid algorithm maps records of produced output table and the original dataset. Thus the incorrect values produced by TKO algorithm exist no more. And also the time of execution required is less, since hybrid algorithm doesn't maintain any temporary table or memory like the TKU algorithm. Thus accurate result can be obtained in moderately less amount of time using the hybrid algorithm.

IV. RESULT AND ANALYSIS

Experiments are done on a personal computer with a configuration: Intel (R) Core (TM) i3-2120 CPU @ 3.30GHz, Windows 7, 4GB memory, Jdk 1.8 and MySQL 5.1 backend database. Book recommendation is web application tool used for designing code in Eclipse and execute on Tomcat server. Three different algorithms i.e. TKO, TKU and hybrid algorithm TKO with TKU are executed separately using the books dataset.

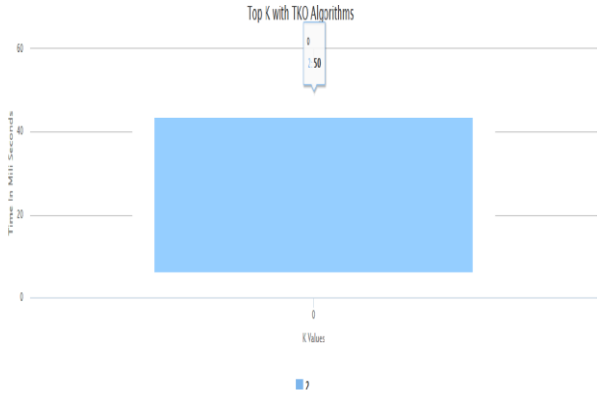


Fig. 3. K value Vs Time (in Milliseconds) result graph for TKO

The result of TKO algorithm i.e. figure 3 shows that for k value of 2 the algorithm requires 50 milliseconds to get executed. Whereas the graph in figure 4 shows that TKU algorithm when executed with similar k value of 2 requires much more time for execution i.e. 51430 milliseconds.

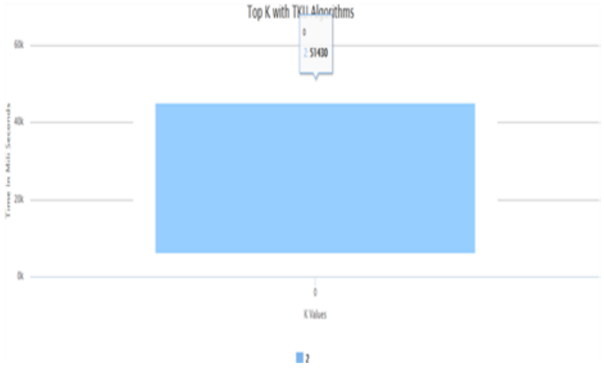


Fig. 4. K value Vs Time (in Milliseconds) result graph for TKU

After execution of the proposed hybrid TKO with TKU algorithm, it can be observed from figure 5 that, hybrid algorithm requires moderately less time for execution i.e. 11259 milliseconds.

Retrieval Number: paper_id//2019©BEIESP
DOI: 10.35940/ijeat.xxxxx.xxxxx

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

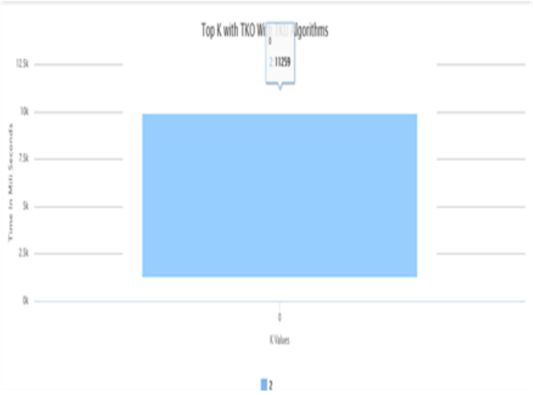


Fig. 5. K value Vs Time (in Milliseconds) result graph for hybrid TKO with TKU

The comparative analysis of all three algorithms can be seen from the figure 6.

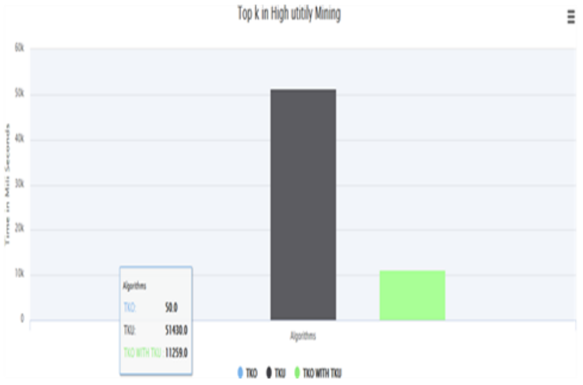


Fig. 6. Comparison between Algorithms

The below table demonstrates the details of above graph:

TABLE III: Comparison Table of algorithms w.r.t Time and K-value

Number	Name of Algorithms	Time In MS	Input (As a K value)
1	TKO	50	2
2	TKU	51430	2
3	TKO WITH TKU	11259	2

V. CONCLUSION

In this paper, the question of, how item-sets which can be of maximum profit to an organization can be gained is addressed using application of book recommendation system where user has to specify k value and the book category. The previously existing algorithms are deeply studied and also implemented. The proposed hybrid algorithm is and implemented for book recommendation system. Theoretical and practical evaluations on different types of real and synthetic data sets display that proposed algorithm performs better than the already existing algorithms with respect to time complexity and accuracy. From above result it can be analyzed that TKO requires less execution time but gives wrong output i.e. garbage values, TKU gives accurate result but requires large execution time. Whereas Hybrid TKO with TKU algorithm requires considerably less execution time and gives accurate result. Hence it can be concluded that, hybrid TKO with TKU algorithm is an efficient algorithm for mining HUI and performs better than TKO and TKU.

REFERENCES

1. Ramah Sivakumar. J.G.R. Sathiaseelan, “A performance based empirical study of the frequent item-set mininf algos.” IEEE International Conf. on Power, Control, Signals and Instrumentation Engg. (ICPCSI-2017).
2. Santi Mariana, Isti Surjandari, Arian Dhini, Asma Rosyidah, Puteri Prameswari, “Association Rule Mining for Building Book Recommendation System in Online Public Access Catalog”, 3rd International Conf. on Science in Information Tech. (ICSITech), IEEE 2017.
3. Pitraji Jomsri, “Book recommendation system for digital library based onm user profile using association rule”, IEEE 2014.
4. Vincent S. Tseng, Philippe Fournier-Viger, Philip S. Yu, Senior Member, Cheng-Wei Wu, Fellow, “Efficient Algorithms for Mining Top-K High Utility Itemsets”, IEEE 2016.
5. Jong Soo Park, serin Lee, “Top k hogh utility item-set mining based on utility-list structure”, IEEE 2016.
6. Y. Li, J. Yeh, and C. Chang, “Isolated items discarding strategy for discovering high utility itemsets”, IEEE 2008.
7. L. Siguenza-Guzman et al., ”Literature Review of Data Mining apps. in Academic Libraries,” The Journal of Librarianship 41, pp. 499-510, 2015.
8. Adinarayana reddy B ,O Srinivasa Rao, MHM Krishna Prasad, ”An Improved UP-Growth High Utility Item-set Mining” Inter- national Journal of Computer apps. (0975-8887) vol. 58-No.2, Nov. 2012
9. Chowdhury Farhan Ahmed, Syed Khairuzzaman Tanbeer, Byeong-SooJeong, and Young-Koo Lee, Member, IEEE ”Efficient Tree Structures for High Utility Pattern Mining in Incremental DB” IEEE Trans. Knowledge and Data Engineering, Vol. 21, Issue 12, pp. 1708-1721, Dec. 2009
10. Han, M. Kamber and J. Pei, Data Mining Concepts and Techniques second edition, San Francisco: Morgan Kaufmann Publishers, 2006
11. Raymond Chan; Qiang Yang; Yi-Dong Shen, ”Mining high utility item-sets” In Proc. of Third IEEE Intl Conf. on Data Mining ,November 2003.
12. P.-N. Tan, M. Steinbach and V. Kumar, Introduction to data mining (Vol. 1), Boston: Pearson Addison Wesley, 2006.
13. P. J. . J. A. M. Azevedo, ”Comparing rule measures for predictive association rules,” In ML: ECML 2007, pp. 510- 517, 2007.
14. P.-N. Tan, M. Steinbach and V. Kumar, Introduction to data mining (Vol. 1), Boston: Pearson Addison Wesley, 2006.

AUTHORS PROFILE

Has pursued B.E computers degree from P.V.P.I.T. SPPU in 2016 and is currently pursuing M.E computers degree from M.M.C.O.E, SPPU. Areas of interest are data mining and big data analytics. Has published “Mining algorithms to archive utility item sets – survey” in International journal of management, technology and engineering, March 2019. This paper focuses on the concept of frequent item-set mining and extends the study further to high utility item-set mining. Also various algorithms with their detailed working are enlisted in this paper. “Intelligent heart attack prediction system using big data” has been published in International journal of recent research in mathematics computer science and information technology, March 2016. This paper focuses on naïve bayes algorithm for the purpose of prediction.

He has pursed Ph.D. from NMIMS University, Mumbai in 2012 and has 40 publications to his credit in International journals and conferences. His areas of interest are big data Analytics, IoT and Security. “Smart garbage monitoring system using internet of things”, “Integrated approach to ontology development methodology with case study”, “Analysis of phonetic matching approaches for Indic languages”, “Ontology supported inference system for Hindi and Marathi”, “Rule-based phonetic matching approach for Hindi and Marathi”, “Survey of data mining with privacy preservation”, “Information retrieval in multilingual environment”, “A review on: advanced artificial neural networks (ANN) approach for IDS by layered method” etc are few of his articles. He is life member of CSI and ISTE and also he is in editorial and review board member of many International Journals.

Retrieval Number: A2043109119/2019©BEIESP
DOI: 10.35940/ijeat.A2043.109119
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication