A Heuristic Approach To Redefine FIS By Matrix Implementation Through Update Apriori 'HuApriori' In Textual Data Set

Neeraj Kumar Verma
Research Scholar Department of Computer Science & Engineering, MUIT Lucknow, India
er.neerajkumar@gmail.com
Dr. Vaishali Singh
Assistant Professor Department of Computer Science & Engineering, MUIT Lucknow, India
singh.vaishali05@gmail.com

ABSTRACT - There are several data mining methods for categorization using association rules nowadays, the most famous of which being the Apriori algorithm. By searching the whole database for k-element frequent item sets, the Apriori method is used to define frequent itemsets from large transactional data sets. According to the Apriori algorithm, we are going to re-evaluate and re-evaluate access time which consumes in scanning the database for k-times looking for k-element frequent item set. In this paper, we will analyse and compare our proposed Updated Hybrid-Apriori Algorithm (HuApriori) with the original Apriori algorithm, which concludes the experimental result to calculate frequent items on several groups of transactions with minimal support (for both Apriori and HuApriori) and improves its performance by reducing the time spent accessing the database by 55%. Our proposed HuApriori algorithm is an enhanced version of Apriori algoritam[4] and working greatly better at each parameter which we include in concluding the results.

Keywords: Apriori, A-Apriori, Minimum Threshold, OLAP, OLTP, Frequent item set, Matrix.

1. INTRODUCTION:

Form the last decade, we experience exponential growth of data surrounding us, from which most of the data in the world was generated over the last two or, three years. This steep growth of data let us alone to think how to handle such massive data which is stored in high volume, variety and viscosity. Another challenge with this data is its usefulness and its utilization as per user demand[6]. By Using some data mining techniques we can successfully handle such challenges by extracting the noble information and important patterns from such huge amount of data for better user-driven decisions.

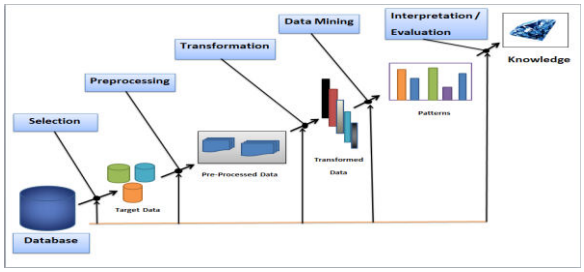

Figure1: Data Mining Step By Step Processing.

The data mining is a youngest interdisciplinary field of computer science which consist of, machine learning, statistics, database systems, visualization, and information sciences. The data mining is a step by step process which starts from the collection of raw data from various resources like files, database, data marts and data warehouse etc[9]. Further it involves pre processing steps like data cleaning, data extraction, data transformation, and up to the final step extracting knowledge. With the help of Knowledge extraction phase we find some meaningful patterns known as knowledge which gives the awesome insights from the data. This whole process is known as Knowledge Discovery in Data (KDD). KDD is an iterative process where evaluation measures can be enhanced, mining can be refined, new data can be integrated and transformed to get different and more appropriate results as per the user demand. The Process of KDD is not only useful in business intelligence but it is also helpful in various other fields like social media, medical, agriculture, social science, educations, etc.
In this paper we analyze a well known Data Mining algorithm called Apriori Algorithm used for Market basket analysis through Frequent item set mining(FIM). We capable to solve various problems through this techniques like data item priority in store, recommendation of item, customer buying pattern analysis etc, but when we talk about the time and space complexity of an algorithm it is very difficult to manage.

The efficiency and throughput of an algorithm is depends on its execution time so that time play a very crucial factor for its success, there are various techniques and algorithms to find the appropriate patterns but the algorithms which take less time to do the same work are being preferred by Researches.

Frequent item set Mining (FIM)

Extracting frequent items from a large transaction data base is major task involves in market basket analysis[13]. Apriori is a popular algorithm for extracting frequent item sets with association rule mining. The Apriori algorithm has been designed to operate on databases containing transactions, such as purchases by customers of a store. An item set is considered as "frequent" if it meets a user-specified support threshold. This minimum support can be set by the user as per their requirement.

For instance, if the support threshold is set to 0.5 (50%), a frequent item set is defined as a set of items that occur together in at least 50% of all transactions in the database.

We have the Association Rule mining which consists of the frequent item set and interesting item sets. Frequent item sets we can calculate using the “if –then rule means if the person buy the bread he also buy the milk”. By Using If- then rule we calculate the item sets by using some threshold value. With help of threshold value we define some frequent item sets. And by using the frequent item set we calculate the interesting item sets. Lets take a Data Set-

(C)

For item=4

Txn	A	B	C	D	E	SUM
T1	1	1	0	0	0	2
T2	0	1	0	1	0	2
T3	0	1	1	0	0	2
T4	1	1	0	1	0	3
T5	1	0	1	0	0	2
T6	0	1	1	0	0	2
T7	1	0	1	0	0	2
T8	1	1	1	0	1	4
T9	1	1	1	0	0	3

(D)

For item=5

Item	COUNT	%	30%	50%	70%
A,B,C,D,E	0	0	N	N	N

(E)

Figure2(A-E): iterations for finding FIS using Min_Supp() Rule

Transactions(Txn)	Items
T1	A,B
T2	B,D
T3	B,C
T4	A,B,D
T5	A,C
T6	B,C
T7	A,C
T8	A,B,C,E
T9	A,B,C

Table1: Data Set

For item=1

Item	COUNT	%	30%	50%	70%
A	6	66.67	Y	Y	N
B	7	77.78	Y	Y	Y
C	6	66.67	Y	Y	N
D	2	22.23	N	N	N
E	1	11.12	N	N	N

(A)

For item=2

Item	COUNT	%	30%	50%	70%
A,B	4	44.45	Y	N	N
A,C	4	44.45	Y	N	N
A,D	1	11.12	N	N	N
A,E	1	11.12	N	N	N
B,C	4	44.45	Y	N	N
B,D	2	22.23	N	N	N
B,E	1	11.12	N	N	N
C,D	0	0	N	N	N
C,E	1	11.12	N	N	N
D,E	0	0	N	N	N

(B)

For item=3

Item	COUNT	%	30%	50%	70%
A,B,C	2	22.23	N	N	N
A,B,D	1	11.12	N	N	N
A,B,E	1	11.12	N	N	N
B,C,D	0	0	N	N	N
B,C,E	1	11.12	N	N	N
C,D,E	0	0	N	N	N

2. LITERATURE REVIEW:

KDD:

The term KDD stands for Knowledge Discovery in Databases, which is specific data mining technique, refers to discovery knowledge from huge data se. The main Objective of KDD is to extract noble information from the number of patterns generated as an output after applying data mining Algorithms on large databases. The Knowledge discovery process is iterative and compress of seven steps. The process is iterative at each stage, implied that moving back to the previous action might be required. The process begins with determining the KDD objectives and ends with the implementation of the discovered knowledge. At that point The loop is closed and the active data mining starts subsequently changes would need to be made in the application domain. Ex:- Offering Various features to the cell phone users in order to reduce churn.

These steps includes (As Shown in Figure 1)-

1. Domain understanding & KDD goals, 2.Selection & Addition, 3.Pre-processing & cleaning, 4.Data Transformation, 5. Data Mining –Prediction & description ,6.Selecting the data mining algorithms,7.Utilizing the data mining algorithm ,8Pattern evaluation,9.Discovery Knowledge(Visualization & Integration).

Data Mining Architecture:

Information mining is the process in which data that changed into previously unknown, which can be probably very beneficial, is extracted from a very sizeable dataset. Records mining architecture or structure of statistics mining strategies are nothing but the various additives which constitute the whole system of data mining.

1. **Sources of Data:-**The region wherein we get our information to paintings upon is called the statistics source or the supply of the information. there are many documentations offered, and one can also argue that the complete global extensive web (WWW) is a big facts warehouse. The information may be everywhere, and a few may live in text documents, a fashionable spreadsheet file, or some other possible source just like the net.

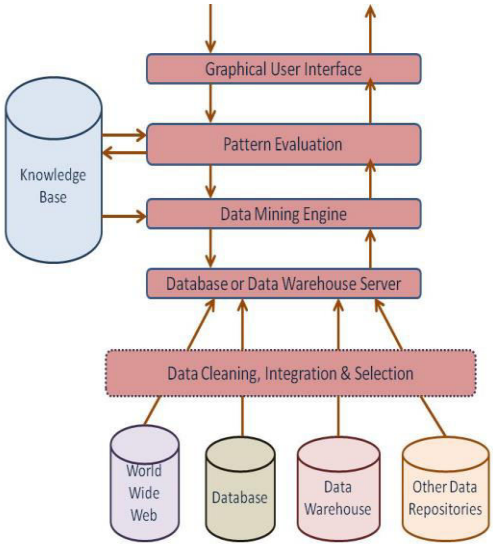

Figure3: Data Mining Architecture

1. **Database or Data Warehouse Server:-**The server is the vicinity that holds all the facts which is prepared to be processed. The fetching of data works upon the consumer's request, and, therefore, the actual datasets can be very non-public.
2. **Data Mining Engine:-**The field of data mining is incomplete without what's arguably the maximum
3. important component of it, called a statistics mining engine. It usually incorporates loads of modules that may be used to carry out a ramification of tasks. The tasks which can be executed can be affiliation, characterization, prediction, clustering, category, and so forth.
4. **Modules for Pattern Evaluation:-**This module of the architecture is specifically hired to degree how interesting the sample that has been devised is in reality. For the assessment purpose, generally, a threshold cost is used. some other important aspect to word here is this module has a direct hyperlink of interplay with the records mining engine, whose fundamental aim is to find exciting patterns.
5. **GUI or Graphical User Interface:-**This module of the architecture is what interacts with the consumer. GUI serves because the a great deal-wanted hyperlink between the consumer and the machine of information mining. GUI's foremost task is to hide the complexities concerning the entire technique of information mining and offer the user with an easy to use and apprehend module which would allow them to get a solution to their queries in an easy to recognize fashion.
6. **Knowledge Base:-**The knowledge base is normally used because the guiding beacon for the sample of the consequences. it might additionally comprise the information from what the users have skilled. The statistics mining engine interacts with the understanding base frequently to both increase the reliability and accuracy of the very last result. Even the sample evaluation module has a hyperlink to the know-how base. It interacts with the knowledge base on a everyday c programming language to get numerous inputs and updates from it.

Mining Association Rule Mining :-Association rule is a basic method of Data mining, which is used to classify the items in the item set as interestingness criteria by applying two key threshold values like lower threshold (Min_Sup()) to decide frequent itemset and upper threshold (Min_Conf()) to decide interesting item set.

Mining Associations rules are basic if-then rule used to classify the item from the item set as per interestingness criteria e.g., "if a person (X) buys a uniform in June then he also buys a school beg". As per the exampleAn association rule R can be defined as-

R1: Buy (X, Uniform) → Buy (X, School_beg)

if a rule R1 is found to be true as per the threshold criteria then the item sets are declared as frequent itemset {Uniform, School_beg}. If rule R1 will be found true in reverse as per threshold criteria like-

R2: Buy (X, School_beg) → Buy (X, Uniform)

then elements of rule R2 are considered as an Interesting item set. The value for Min_Supp () and Min_Conf () is defined by the use

r which represents the constraints for the rules, the value of support and confidence must be optimal and point of analysis because if the value of support is too low then an unwanted set of items will be included into the candidate subset in various n-element item set and if the support is too high then some frequent set of items are missing due to failed in interestingness criteria (Lower Threshold) and included to a discarded subset which will affect the analysis[4]. The research behind the association rule is made motivated towards real-life applications like e-commerce, banking, healthcare, and manufacturing, etc. to make it more successful.

1. APRIORI ALGORITHAM

Apriori is a classical algorithm that is used to mine frequent item sets to derive various association rules[5]. Association mining is a technique that can discover interesting relationships hidden in transaction datasets. This approach first finds all frequent item sets and generates strong association rules from frequent item sets.

Apriori is the most well-known association mining algorithm which identifies frequent individual items first and then performs a breadth-first search strategy to extend individual items to larger itemsets until larger frequent itemsets cannot be found. Apriori set of rules most widely time-venerated, simple, and clean to An executable set of rules that is used to mining all object Devices referred to as commonplace itemsets in the database as in step with standards Described. The set of guidelines is used to find out common itemsets via the use of Searching the database. To do the identical it applies Constraints referred to as decrease threshold (min_supp()) that may be a Possibility of appearing the facts items a number of the datasets And the better threshold used a conditional possibility which Makes high quality the statistics gadgets seem from decrease threshold Supposed to be proper.

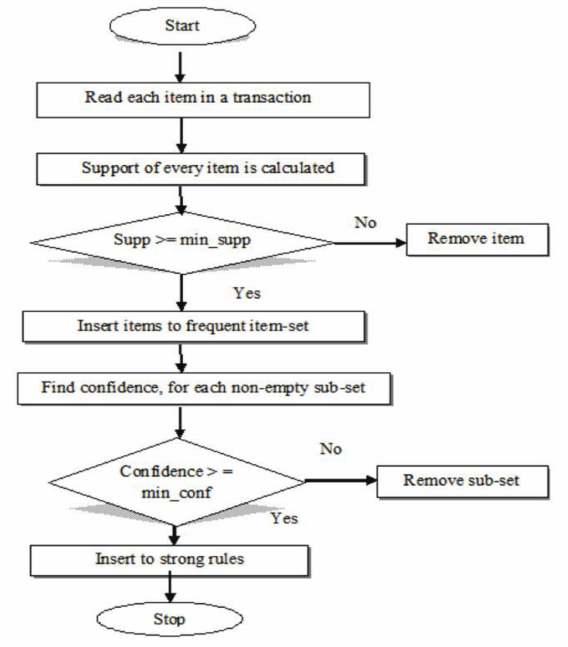

Figure4: Apriori Flowchart

The Apriori algorithm defines in following steps-

Step 1: Define –

Item set I:={ I1, I2, I3…….. Ii} ($\forall$ i $\geq$ 1)

Transaction set T: = {T1, T2, T3……Tm} ($\forall$ m $\geq$ 1)

k-element item set: = {I1, I2, I3….In} ($\forall$ n $\geq$ 1)

//n- element item set is a set belongs to any transaction T_k from T & n-element item set I.

L_n: = Labelled candidate subset

Step 2: Let α is minimum support of item set which defines the frequency of occurrence of data items in a transaction set of transactions.

Step 3: To define frequent item sets among transactions create two selections subsets first called Candidate subset (CSS) as per the successful criteria of α and second for unsuccessful criteria of α, called Discarded Subset (DSS).

Step 4: Let CSS_n is a candidate subset of size k and L_n is a labeled candidate subset of frequent itemset of size n along with their transaction occurrence.

3. Proposed Algorithm For Defining Improved Mining Association Rule (HuApriori)

New Proposed Updated Apriori used Matrix implementation of data set for finding the Max number of items grouped in any single transaction as per the flow chart given below-

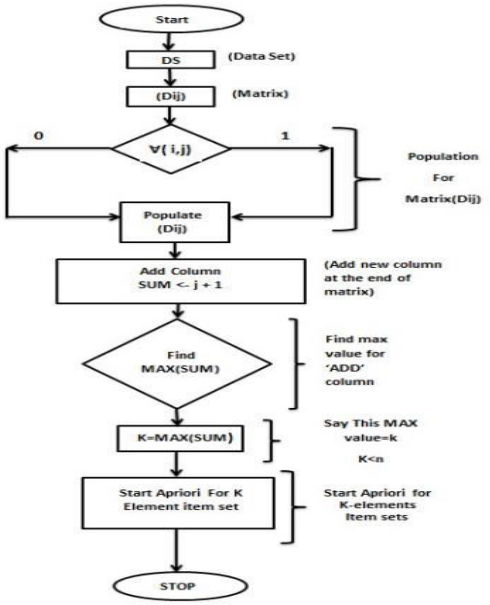

Figure5: Flow Chart of Updated Apriori(HuApriori) algorithm

4. An Execution Of HuApriori Algorithm

Hear we take four Data set from open source platform, DS-1with the statistics among one contains 22039 number of Transactions with 4215 number of items , DS-2 contains 15000 No of Transactions with 4089 number of items, DS-3 contains 9000 No of Transactions with 3922 number of items and DS-4 contains 4000 No of Transactions with 3641 number of items as shown in Table-3, which was executed in Python with the same environment of intel 5i eight generation, 8GB RAM & 1TB HDD as shown in Table-3.

Data Set	No of Items(I_i) & No of Transactions(Txn)	Min_Sup() in %age	Total No.of items (n)	Max No. of Items appear in	Execution Time-Apriori(T_A) in MilSec	Execution Time-Apriori(T_{HA}) in MiliSec	Execution Time Difference-(T_A-T_{HA})	Execution Rate(%)
DS-1	Ii=4215 & No of Txn=22039	30	4215	1117	6159.352779	2775.66433	3383.68845	45
		50	4215	1117	2755.106211	2727.006912	28.09929848	99
		70	4215	1117	2712.578297	2704.259634	8.318662643	100
DS-2	Ii=4089 & No of Txn=15000	30	4089	573	4084.969997	1839.228153	2245.741844	45
		50	4089	573	1894.00959	1798.013687	95.99590302	95
		70	4089	573	1813.723326	1724.029303	89.69402313	95
DS-3	Ii=3922 & No of Txn=9000	30	3922	753	2192.187548	1071.732759	1120.454788	49
		50	3922	753	1406.403303	1016.978264	389.4250393	72
		70	3922	753	1018.704653	981.4305305	37.27412224	96
DS-4	Ii=3641 & No of Txn=4000	30	3461	680	888.2453442	418.8838005	469.3615437	47
		50	3461	680	422.4140644	418.1544781	4.259586334	99
		70	3461	680	411.42869	406.4588547	4.969835281	99

Table-3: Apriori v/s HuApriori as per rate of Execution Time

5. Result Analysis of HuApriori

Now, as per the statistics, as shown in Table-4. We can easily be find out that in data set DS-1 which contains total number of 22039 transactions with 4215 number of items.

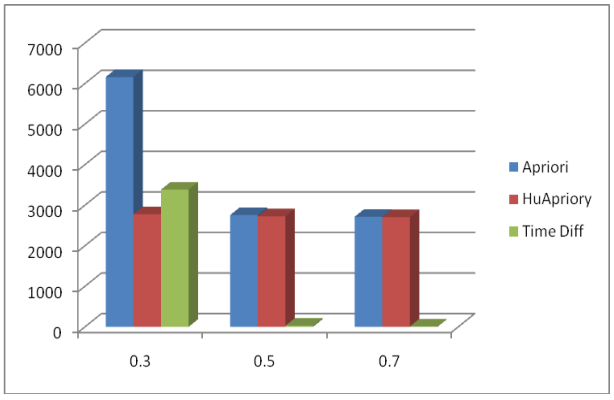

Figure6**:** Apriori v/s HuApriori as per reduced access time for DS-1.

Now we execute both like Apriori and UpApriory with the data set DS-1with minimum support value in the range of 30%, 50% and 70% and get execution time for Apriori and UpApriory is 6159.35 and 3383.67, 2755.10 and 2727.00 & 2712.58 and 2704.26 Millisecond respectively. It can easily be observed that for Min_Supp(30%) HuApriori is 55% faster then Apriori as shown in Fugure-6.

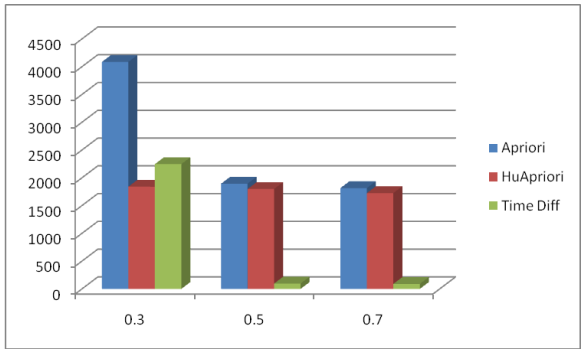

Figure-7: Apriori v/s HuApriori as per reduced access time for DS-2.

As shown in Fugure-7, for data set DS-2 which contains total number of 15000 transactions with 4089 number of items. Now we execute both like Apriori and UpApriory with the data set DS-1with minimum support value in the range of 30%, 50% and 70% and get execution time for Apriori and UpApriory is 4084.97 and 1839.22, 1894.00 and 1798.01& 1813.72 and 1724.03 Millisecond respectively. It can easily be observed that for Min_Supp(30%) HuApriori is again 55 faster then Apriori, for Min_Supp(50%) HuApriori is 5% faster then Apriori, for Min_Supp(70%) HuApriori is again 5 faster then Apriori.

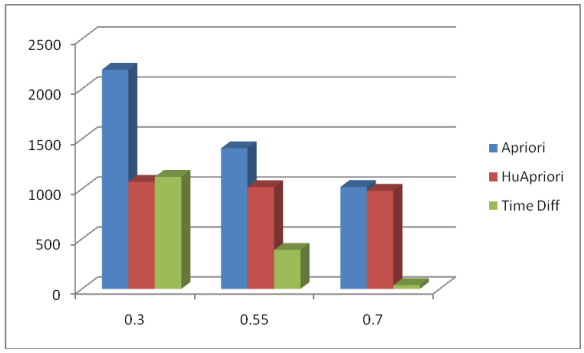

Figure-8: Apriori v/s HuApriori as per reduced access time for DS-3

for data set DS-3 which contains total number of 9000 transactions with 3932 number of items, with minimum support value of 30%, 50% and 70%, get execution time for Apriori and UpApriory is 2192.18 and 1406.40, 1018.70 and 1071.73 & 1016.97 and 981.43 Millisecond respectively. Which predict observations for Min_Supp(30%) HuApriori is again 51 faster then Apriori, for Min_Supp(50%) HuApriori is 28% faster then Apriori, for Min_Supp(70%) HuApriori is again 4 faster then Apriori.

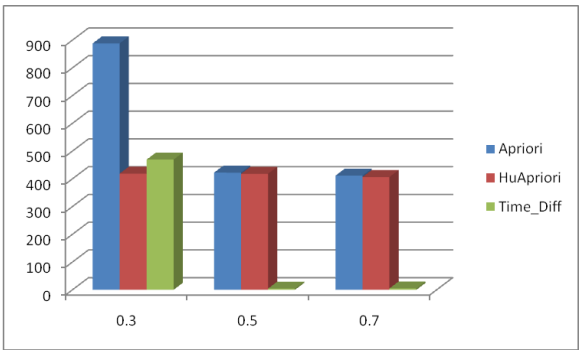

Figure-9: Apriori v/s HuApriori as per reduced access time for DS-4.

for DS-4 which contains total number of 4000 transactions with 3641 number of items, with minimum support value of 30%, 50% and 70%, get execution time for Apriori and UpApriory is 888.24 and 422.41, 411.42 and 418.88 & 418.15 and 406.45 Millisecond respectively. Which predict observations for Min_Supp(30%) HuApriori is again 51 faster then Apriori, for Min_Supp(50%) HuApriori is 28% faster then Apriori, for Min_Supp(70%) HuApriori is again 4 faster then Apriori.

6. Conclusion

In this paper, we conclude to enhance the execution rate by reducing the number of scanning of transactions by eliminating such iterations which have items greater then maximum number of items in any single transaction in the data set define as MAX (SUM).

As per the statistics from execution state that HuApriori is faster 55% & 1% then Apriori for Min_Supp(30) & Min_Supp(50) in DS-1 as shown in Figure-6, and 55%, 5% & 5% faster for Min_Supp(30), Min_Supp(50) & Min_Supp(70) in DS-2 as shown in Figure-7, and 51%, 28% & 4% faster for Min_Supp(30), Min_Supp(50) & Min_Supp(70) in DS-3 as shown in Figure-8, and finally 53%, 1% & 1% faster for Min_Supp(30), Min_Supp(50) & Min_Supp(70) in DS-3 as shown in Figure-9

All of the above statistics show that our proposed HuApriori algorithm is an improved version of the Apriori algorithm[4], and that it performs much better at every parameter that we include in concluding the results, as the transaction set grows larger and larger with uniform and variable Minimum Support for transactions in the Transaction Set.

7. References

[1] S. Rao, R. Gupta, "Implementing Improved Algorithm Over APRIORI Data Mining Association Rule Algorithm", *International Journal of Computer Science And Technology*, pp. 489-493, Mar. 2012

[2] F. Crespo and R. Weber, "A methodology for dynamic data mining based on fuzzy clustering," *Fuzzy Sets and Systems*, vol. 150, no. 2, pp. 267–284, Mar. 2005.

[3] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery in databases," *AI Magazine*, vol. 17, no. 3, p. 37, 1996.

[4] J. Han, M. Kamber, "Data Mining: Concepts and Techniques*", Morgan Kaufmann Publishers*, Book, 2000.

[5] X. Wu, V. Kumar, J. Ross Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, P. S. Yu, Z.-H. Zhou, M. Steinbach, D. J. Hand, and D. Steinberg, "Top 10 algorithms in data mining," *Knowledge and Information Systems*, vol. 14, no. 1, pp. 1–37, Dec. 2007.

[6] R. Srikant, "Fast algorithms for mining association rules and sequential patterns," *University Of Wisconsin*, 1996.

[7] H. H. O. Nasereddin, "Stream data mining," *International Journal of Web Applications*, vol. 1, no. 4, pp. 183–190, 2009.

[8] R. Agrawal, T. Imielinski, and A. Swami, "Mining association rules between sets of items in large databases," in *ACM SIGMOD Record*, vol. 22, pp. 207–216, 1993

[9] M. Halkidi, "Quality assessment and uncertainty handling in the data mining process," in Proc, EDBT Conference, Konstanz, Germany, 2000.

[10] F. H. AL-Zawaidah, Y. H. Jbara, and A. L. Marwan, "An Improved Algorithm for Mining Association Rules in Large Databases," Vol. 1, No. 7, 311-316, 2011

[11] T. C. Corporation, "Introduction to Data Mining and Knowledge Discovery*", Two Crows Corporation*, Book, 1999.

[12] Neeraj kumar Verma, Sandeep Kumar"An Alternate Approach to Improve Access Time for Defining Frequent Item Set Through 'A-Apriori' In Textual Data Set" 2020 5th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE), DOI: 10.1109/ICRAIE51050.2020.9358344

[13] B. Moghe , H. K. Pamnani , Neeraj Kumar Verma "A Comparative Review of Various Data Mining Algorithms for Customer Behaviour Identification Using Market Basket Analysis" Journal of Data Mining and Management, Volume-6, Issue-1 (January-April, 2021), e-ISSN: 2456-9437.