International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 7, Issue 7, July 2017)

Performance Analysis of Frequent Itemsets Mining Algorithms and Association Rule Mining for Efficient Data Mining

Manoj Niwariya[1], Piyush K. Dubey[2]

[1]*Asst. Professor, Department of Computer Application, Makhanlal Chaturvedi National University of Journalism and Communication (MCNUJC), Bhopal, (M.P)*

[2]*Asst.Professor & IT Officer, Department Of Computer Application & Science, Shri L.B.S. College Ganj Basoda, Dist-Vidisha (M.P.)*

***Abstract* - Mining frequent itemsets is the very crucial task to find the association rules between the various items. Investigations of Frequent Itemset (or pattern) Mining is recognized in the field of data mining in light of its expansive applications in mining association rules, connections, and graph pattern imperative in light of incessant patterns, successive patterns, and numerous other data mining assignments. Efficient algorithms for mining successive itemsets are critical for mining association guidelines and in addition for some other data mining tasks. The major challenge found in frequent pattern mining is a large number of result patterns. The problem of mining frequent itemsets arises in the large transactional databases when there is need to find the association rules. Various algorithms have been analyzed in this paper these different algorithms have strengths and weakness in different type of datasets. As a measure of execution for the most part the normal number of operations or the normal execution times of these algorithms have been examined.**

***Keywords:-* Frequent Itemsets Mining, Association Rule Mining, Data Mining.**

I. INTRODUCTION

Data mining is an as of late developing field, interfacing the three worlds of databases, artificial intelligence and insights (Lindell 2002). It includes the utilization of data investigation apparatuses to find beforehand obscure, legitimate patterns and connections in substantial datasets (Seifert 2004). Display made for data mining can be prescient or expressive. Prescient models make an expectation about estimations of data utilizing known outcomes found from various data. Spellbinding models distinguish patterns of connections in data. Normal tasks of prescient models are classification, regression, time series examination and expectation. Clustering, outline, affiliation guidelines and succession revelation are normal undertakings of prescient data mining models (Dunham 2002). These are portrayed in Fig. 1.

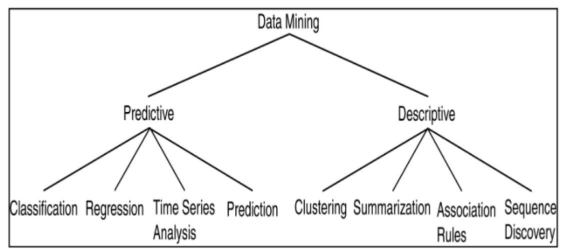

Fig. 1. Data mining models and tasks

There are essentially three data mining methods: classification, clustering and association rule mining. Classification utilizes a preparation set and manufactures a classifier to anticipate the classes of new examples. Clustering separates dataset into groups of which individuals are like each other and not quite the same as individuals from different bunches. Association run mining discovers patterns and connections among dataset. These systems are quickly presented in following subsections.

Data mining, characterized as the way toward finding information or patterns from enormous measures of data (Liu 2009), has turned into a well known approach to find vital learning. Direct mail promoting, site personalization; bioinformatics, charge card extortion recognition, text examination and market bushel investigation are a few illustrations where data mining strategies are ordinarily utilized. Data mining models are separated into two as prescient and engaging. Prescient models incorporate undertakings relapse, classification, time arrangement examination and forecast. Unmistakable models incorporate undertakings clustering, outline, association rules and arrangement revelation (Dunham 2002). Association rule mining uncovers connections among set of things in a database in two stages visit itemset mining and creating association rules from these itemsets. It was right off the bat presented by (Agrawal 1993), trailed by well known Apriori algorithm (Agrawal 1994) which recorded in top ten data mining algorithms (Wu 2008).

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 7, Issue 7, July 2017)

In spite of the fact that it supported data mining research, Apriori algorithm has a bottleneck of numerous database check for hopeful itemset era. In (Han 2000) FP-Growth algorithm proposed for visit itemset mining without applicant era. It stores data of database in tree structure called FP-tree and sweeps database just twice. Later in (Pavon 2006), Matrix Apriori algorithm is proposed. It is like FP-Growth in the method for database checking and putting away data of database in a reduced data structure however network data structure is utilized rather than tree. Data mining is effectively connected to many fields like clustering in bioinformatics, association rules in showcase crate examination, classification in credit scoring, time arrangement investigation in money related choice supporting. In any case, the expanding energy of PCs taking care of enormous sum data and noxious utilization made data mining a hazard to protection of people and organizations.

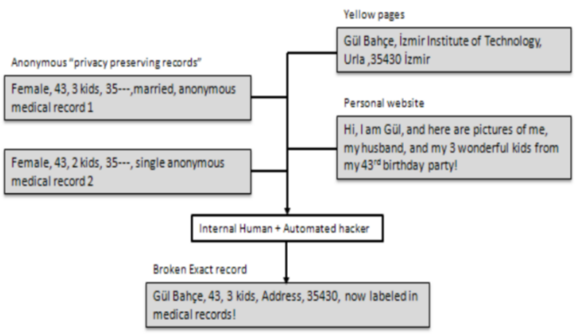

Figure 2. Privacy problem example

In Figure 1.1 a straightforward case of security issue caused by consolidating data from various destinations is given. Postal divisions of medicinal records are anonymized to ensure revelation of patient and data in individual site and address in business repository don't cause a protection issue exclusively. Be that as it may, a macilious inner human and programmer may consolidate the data in various locales and mark restorative record of patient.

Public sensitivity against data mining expanded on the grounds that it is seen a danger to people private data as appeared in the case above. Then again, data digging is essential for proficiently finding learning. Protection safeguarding data mining emerge from the requirement for keep performing data mining proficiently yet saving private data or learning of people and organizations.

It is characterized as data mining systems that utilization particular ways to deal with secure against the exposure of private data may include anonymizing private data, misshaping touchy esteems, encoding data, or different intends to guarantee that delicate data is ensured (Liu 2009). Protection safeguarding data mining is partitioned into two noteworthy classes: data covering up and rule stowing away. Data concealing means to outline new conventions to irritate, anonymize or scramble crude data while delicate private data is secured and basic patterns can even now be found (Subramanian 2008). Rule concealing alludes to outline algorithms is such a path, to the point that delicate rules or patterns remain unrevealed while remaining rules or patterns can at present be mined. The first data is contorted or hindered by rule concealing algorithms. Security, the new bearing of data mining research is the fundamental inspiration for begin purpose of this exploration. It is chosen to apply protection saving data digging systems for visit itemset mining. Looking over writing, it has been seen that numerous algorithms for association rule or incessant itemset covering up are Apriori based and as it is specified above it has a weakness of various database checking. In this way, algorithms without applicant era are considered right off the bat. Grid Apriori and FPGrowth algorithms are analyzed and a paper arranged (Yıldız 2010) from this initially period of research. Since comes about demonstrated that Matrix Apriori performed better and its grid data structure is anything but difficult to deal with, investigate consider is coordinated to proposing an incessant itemset concealing algorithm in view of Matrix Apriori. As its lattice data structure gives design data, the algorithm is adjusted to have itemset concealing capacities. Also, imaginative heuristics for choice of thing contortion are proposed which utilize design length rather exchange length proposed by past examinations on visit itemset covering up. These algorithms are looked at for changed cases and results talked about. Another paper for the second period of the examination has been arranged and submitted. All the advance is delineated in this examination. The objective and the structure of this examination think about are given in next subsections.

II. Association Rule Mining

Association rule miming discovers connections and patterns between things in a database. It is a two stage prepare. Right off the bat, visit itemsets are found and furthermore from these itemsets, rules are created.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 7, Issue 7, July 2017)

Formal meaning of association rule mining is Given a set of items $I = \{I1, I2, \ldots, Im\}$ and a database of transactions $D = \{t1, t2, \ldots, tn\}$ where $ti = \{Ii1, Ii2, \ldots, Iik\}$ and $Iij \in I$ and X,Y are set of items, the association rule problem is to identify all association rules $X \rightarrow Y$ with a minimum support and confidence where support of association rule X → Y is the percentage of transactions in the database that contain $X\ U\ Y$ and confidence is the ratio of support of X U Y to support of X. Simply the acquiring of one item when another item is bought in the market bushel data speaks to an association rule. A notable illustrative case of association rules is "Diaper! Brew" which can be clarified by the way that, when fathers purchase diapers for their children, they additionally purchase lager in the meantime for their end of the week's diversion observing (Liu 2008). Some popular association rule mining algorithms Apriori (Agrawal 1994), ECLAT (Zaki 1997) and FP-Growth (Han 2000).

III. Frequent Itemset Mining

The advance in bar-code and computer innovation has made it conceivable to gather data about deals and store as exchanges which is called wicker called basket data. This stored data attracted in researchers to apply data mining to wicker container data. Subsequently association rules mining became a force to be reckoned with which is said as synonymous to advertise wicker container investigation. As expressed before association rule mining is a two stage prepare. Right off the bat, visit itemsets are discovered utilizing least help esteem, and this progression is the primary convergence of association rule mining algorithms. Later from these itemsets utilizing least certainty esteem rules are created. As the varying piece of the algorithms are visit itemset discovering part, association rule mining, visit itemset mining or continuous example mining terms are utilized conversely.

As frequent data itemsets mining are vital in mining the Association rules. In this manner there are different strategies are proposed for producing continuous itemsets with the goal that association rules are mined effectively. The methodologies of creating regular itemsets are partitioned into essential three systems.

- Horizontal layout based data mining techniques o Apriori algorithm o DHP algorithm o Partition o Sample o A new improved Apriori algorithm.
- Vertical layout based data mining techniques o Eclat algorithm.
- Projected database based data mining techniques o FP-tree algorithm o H-mine algorithm.

There are many algorithms used to mine continuous itemsets. Some of them, exceptionally surely understood, began a radical new period in data mining. They made the idea of mining successive itemsets and association rules conceivable. Others are varieties that acquire enhancements fundamentally terms of preparing time. The absolute most imperative algorithms quickly clarified in this report. The algorithms shift predominantly in how the hopeful itemsets are produced and how the backings for the applicant itemsets are numbered. This area will present some illustrative algorithms of mining association rules and regular itemsets.

Apriori Algorithm

The to start with algorithm for mining all continuous itemsets and solid association rules was the AIS algorithm by [3]. Not long after that, the algorithm was enhanced and renamed Apriori. Apriori algorithm is, the most traditional and critical algorithm for mining regular itemsets. Apriori is utilized to locate all regular itemsets in a given database DB. The key thought of Apriori algorithm is to make various disregards the database. It utilizes an iterative approach known as a broadness initially seek (level-wise pursuit) through the hunt space, where k-itemsets are utilized to explore (k+1)-itemsets.

The working of Apriori algorithm is reasonably relies on the Apriori property which expresses that" All nonempty subsets of a continuous itemsets must be visit" [2]. It additionally depicted the counter monotonic property which says if the framework can't breeze through the base help test, all its supersets will neglect to finish the test [2, 3]. Along these lines if the one set is occasional then all its supersets are additionally continuous and the other way around. This property is used to prune the infrequent candidate elements. In the beginning, the set of frequent 1-itemsets is found. The set of that contains one item, which fulfill the support threshold, is signified by L_1. In each ensuing pass, we start with a seed set of itemsets observed to be substantial in the past pass. This seed set is utilized for creating new possibly vast itemsets, called applicant itemsets, and check the real help for these competitor itemsets amid the disregard the data. Toward the finish of the pass, we determine which of the competitor itemsets are in reality substantial (incessant), and they turn into the seed for the following pass. In this manner, L_1 is utilized to discover L_2, the arrangement of incessant 2-itemsets, which is utilized to discover L_1 , et cetera, until not any more successive k-itemsets can be found. The component initially concocted by [2] in Apriori algorithm is utilized by the numerous algorithms for visit design era.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 7, Issue 7, July 2017)

The essential strides to mine the incessant components are as per the following [3]:

Generate and test: In this first find the 1-itemset frequent elements L_1 by scanning the database and removing all those elements from C_1 which cannot satisfy the minimum support criteria.

• *Join step:* To attain the next level elements Ck join the previous frequent elements by self join i.e. L_{k-1} L_{k-1} known as Cartesian product of L_{k-1}. i.e. This step generates new candidate k-itemsets based on joining L_{k-1} with itself which is found in the previous iteration. Let Ck denote candidate k-itemset and Lk be the frequent k-itemset.

• *Prune step:* Ck is the superset of Lk so members of Ck may or may not be frequent but all K - 1 frequent itemsets are included in Ck thus prunes the Ck to fnd K frequent itemsets with the help of Apriori property. i.e. This progression wipes out a portion of the competitor k-itemsets utilizing the Apriori property A sweep of the database to determine the include of every applicant Ck would bring about the determination of Lk (i.e., all hopefuls having a check no not as much as the base help number are visit by definition, and subsequently belong to Lk). Ck, be that as it may, can be tremendous, thus this could include grave calculation. To recoil the extent of Ck, the Apriori property is utilized as takes after. Any (k-1)-itemset that is not visit can't be a subset of a regular k-itemset. Thus, assuming any (k-1)- subset of applicant k-itemset is not in Lk-1 then the competitor can't be visit either thus can be expelled from C#. Step 2 and 3 is rehashed until the point that no new hopeful set is created.

To illustrate this, suppose n frequent 1-itemsets and minimum support is 1 then according to Apriori will generate n^2 +(n 2)candidate 2 - itemset (n 3)candidate 3 - itemset and so on. The total number of candidates generated is greater than $\sum_{k-1}^{n}(n\ k)$ in this way assume there are 1000 components then 1499500 competitors are created in 2 itemset visit and 166167000 are delivered in 3-itemset visit [11]. It is undoubtedly Apriori algorithm effectively finds the continuous components from the database. In any case, as the dimensionality of the database increment with the quantity of things at that point:

• More search space is needed and I/O cost will increase.

• Number of database scan is increased thus candidate generation will increase results in increase in computational cost. In this manner numerous varieties have been happens in the Apriori algorithm to limit the above restrictions emerges because of increment in size of database.

These in this way proposed algorithms receive comparable database examine level by level as in Apriori algorithm, while the techniques for applicant era and pruning, bolster numbering and competitor portrayal may vary. The algorithms enhance the Apriori algorithms by:

- Reduce passes of transaction database scans
- Shrink number of candidates
- Facilitate support counting of candidates These algorithms are as follows:

Direct Hashing and Pruning (DHP):

It is assimilated that lessening the competitor things from the database is one of the critical task for expanding the effectiveness. In this manner a DHP system was proposed [7] to decrease the quantity of applicants in the early passes C# for k 9 1 and subsequently the extent of database. In this strategy, bolster is numbered by mapping the things from the competitor list into the basins which is separated by help known as Hash table structure. As the new itemset is encountered if item exist earlier then increase the bucket count else insert into new bucket. Thus in the end the bucket whose support count is less the minimum support is removed from the candidate set. In this way it reduce the generation of candidate sets in the earlier stages but as the level increase the size of bucket also increase thus difficult to manage hash table as well candidate set.

Partitioning Algorithm:

Partitioning algorithm [1] is based to locate the regular components on the premise parceling of database in n parts. It conquers the memory issue for vast database which don't fit into primary memory since little parts of database effectively fit into fundamental memory. This algorithm divides into two passes,

1. In the first pass whole database is divided into n number of parts.
2. Each partitioned database is loaded into main memory one by one and local frequent elements are found.
3. Combine the all locally frequent elements and make it globally candidate set.
4. Find the globally frequent elements from this candidate set. It ought to be noticed that if the base help for exchanges in entire database is min_sup then the base help for apportioned exchanges is min-sup number of exchange in that parcel. A neighborhood visit itemset might be visit as for the whole database in this way any itemset which is conceivably visit must incorporate into any of the continuous segment.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 7, Issue 7, July 2017)

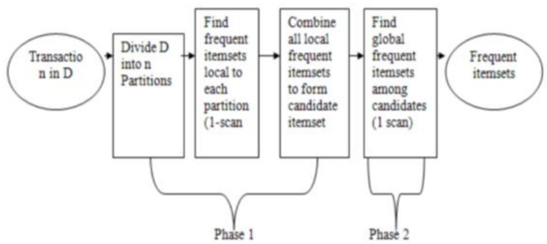

Figure 2-1: Mining Frequent itemsets using Partition algorithm [13]

As this algorithm ready to decrease the database filter for producing continuous itemsets however now and again, the time expected to register the recurrence of competitor creates in each parcels is more noteworthy than the database check along these lines brings about expanded computational cost.

Sampling Algorithm:

This algorithm [10] is utilized to beat the constraint of I/O overhead by not considering the entire database for checking the recurrence. It is quite recently situated in the thought to pick an irregular example of itemset R from the database rather than entire database D. The example is picked such that entire specimen is obliged in the principle memory. Along these lines we attempt to locate the regular components for the specimen just and there is opportunity to miss the worldwide continuous components in that example in this way bring down limit bolster is utilized rather than real least help to locate the successive components neighborhood to test. In the best case just a single pass is expected to locate every single successive component if every one of the components incorporated into test and if components missed in test at that point second pass are expected to discover the itemsets missed in first pass or in test [13]. Along these lines this approach is useful if proficiency is more critical than the precision since this approach gives the outcome in less output or time and defeat the confinement of memory utilization emerges because of era of expansive measure of datasets yet comes about are not as much exact.

Dynamic Itemset Counting (DIC):

This algorithm [4] likewise used to diminish the quantity of database filter. It depends on the descending revelation property in which includes the hopeful itemsets at various purpose of time amid the sweep. In this unique pieces are shaped from the database set apart by begin focuses and not at all like the past strategies of Apriori it powerfully changes the arrangements of hopefuls amid the database filter.

Not at all like the Apriori it can't begin the following level output toward the finish of first level sweep, it begin the sweep by beginning mark joined to every dynamic parcel of hopeful sets. Along these lines it decrease the database examine for finding the successive itemsets by simply including the new applicant anytime of time amid the run time. Be that as it may, it produces the expansive number of hopefuls and figuring their frequencies are the bottleneck of execution while the database examines just take a little piece of runtime. Suspicion [12, 13]: The execution of all the above algorithms depends on a verifiable presumption that the database is homogenous and along these lines they won't create excessively numerous additional competitors than Apriori algorithm does. For instance, if all segments in Partition algorithm are not homogenous and about totally extraordinary arrangements of nearby continuous itemsets are produced from them, the execution can't be great.

Improved Apriori algorithm:

It was absorbed in [15] [13] that the enhanced algorithm depends on the blend of forward output and turn around sweep of a given database. On the off chance that specific conditions are fulfilled, the enhanced algorithm can extraordinarily diminish the cycle, examining times required for the revelation of hopeful itemsets. Assume the itemset is visit, the majority of its nonempty subsets are visit, opposing to the given condition that one nonempty subset is not visit, the itemset is not visit. In light of this idea, proposes an enhanced technique by joining forward and switch considering: locate the greatest continuous itemsets from the most extreme itemset right off the bat, at that point, get all the nonempty subsets of the incessant itemset. We know they are visit on the premise of Apriori's property. Furthermore, filter the database again from the most reduced itemset and check the successive. Amid this filtering, in the event that one thing is discovered being rejected in the continuous set, it will be prepared to judge whether the itemsets related with it is visit, on the off chance that they are visit, they will be included the barrel-structure (incorporate regular itemsets).we get all the incessant itemsets. The key of this algorithm is to locate the most extreme successive itemset quick.

Advantage:

• According to [15] The consumed time of Apriori and the improved algorithm is:

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 7, Issue 7, July 2017)

Table 2-2:
Comparison of aprori and improved Apriori [16]

Algorithm	Time
Apriori	23 min
Improved Algorithm	10 min

This algorithm gets the maximum frequent itemsets directly, then, get their subsets and compare them with the items in the database. Thus, it saves much time and the storing space.

Disadvantages:

- It will lose mean if the maximum frequent cannot be found fast.
- It cannot fit the situation that if there are still many items not included in the frequent set consisted of the maximum frequent itemsets and all of their nonempty subsets.

IV. Literature Review

A. Kaur, V. Aggarwal and S. K. Shankar [6] have introduced the most well-known issues in data mining is to discover visit itemsets. There are different algorithms which concentrate such itemsets from vast database in light of Minimum Support Threshold (MST). They additionally create association rules in light of Minimum Confidence Threshold (MCT). These two limit esteems are characterized by client or association. Apriori is a standout amongst the most famous data mining algorithms yet it creates all regular itemsets and association rules which might be of client's advantage or may not be. Proposed algorithm prunes all uninteresting continuous itemsets produced in each level and just considers those things and rules which are of intrigue in light of MST and MCT. It spares significant storage room and time. Each level prunes such things and rules which are of no intrigue and advances the resultant rundown to next cycle.

B. M. Al-Maqaleh and S. K. Shaab [7] proposed the Identifying regular thing sets is a standout amongst the most essential issues confronted by the learning revelation and data mining group. There have been various fruitful algorithms created for separating incessant thing sets in expansive databases. Visit thing set mining prompts the revelation of associations and connections among things in vast value-based or social datasets. An issue with such a procedure is, to the point that the arrangement of intriguing patterns must be performed just on visit thing sets. Pushing imperatives in visit thing sets mining can enable pruning the hunt to space.

In this paper, a proficient algorithm is proposed to incorporate certainty measure amid the way toward mining continuous thing sets, which produces sure incessant thing sets. Subsequently, the proposed algorithm produces solid association rules from these certain regular thing sets. This procedure has been actualized and the exploratory outcomes demonstrate the convenience and adequacy of the proposed algorithm.

P. Dong and B. Chen [8] Discovered maximumextreme incessant thing sets is a key issue in data mining. With a specific end goal to defeat the lacks of apriori-like algorithms which receive hopeful itemsets era and-test approach, we propose another algorithm ML_DMFIA which in view of DMFIA to mine most extreme continuous itemsets in different level association rules. ML_DMFIA uses FP-tree structure and up-down dynamic developing seeking thought which can abstain from making numerous ignores database and does not create hopeful itemsets, subsequently, it lessens CPU time and I/O time astoundingly. Our execution think about demonstrates that ML_DMFIA is more productive than ML_T2 algorithm for mining both long and short incessant itemsets in mining different level association rules.

P. Dong and B. Chen [9] displayed an examination consider in which the most extreme continuous itemsets is a key issue in data mining applications. A large portion of the past examinations embrace an Apriori-like applicant itemsets era and-test approach, in any case, competitor itemsets era is exorbitant. In this investigation, we propose another algorithm named ML_Pincer for finding most extreme successive itemsets in different level association rules. ML_Pincer algorithm consolidates the best down and the base up bearings dynamic extending seeking thoughts, additionally, it utilizes two-way pruning strategy: the data which accumulated one way can prune more applicant itemsets amid the hunt the other way. It diminishes hopeful itemsets significantly and abstains from making numerous disregards database, subsequently, it decreases CPU time and I/O time surprisingly. Analyses demonstrate that ML_Pincer algorithm is more proficient than PMAM algorithm, particularly when some greatest regular itemsets are long.

Pei-Qi Liu, Zeng-Zhi Li and Yin-Liang Zhao [10] portrayed the productivity of mining association rules is a critical field of KDD. The algorithm Apriori is an established algorithm in mining association rules. It is an expansiveness initially seek on the grid space of itemsets.

In spite of the fact that it makes utilization of hostile to monotone of itemsets to diminish seeking expansiveness, the algorithmic unpredictability of time is as yet the exponential amount. The ideas of the era and the ordinal itemsets tree are presented. The ordinal itemsets tree is the dynamic portrayal of mining connection of itemsets, and the vegetal capacity of the ordinal itemsets tree is depicted by the era. Through the investigation of the association rules, the conclusion that all successive itemsets are not all vegetal itemsets and all vegetal itemsets are largely visit itemsets is found. With this conclusion, the quantity of the applicant itemsets can be decreased further to enhance the productivity of mining association rules and diminish the seeking expansiveness. As indicated by the era, the AprioriFREQ algorithm, which is the change algorithm of Apriori, is composed in this article. By testing, the efficiency of the AprioriFREQ algorithm is obviously higher than the Apriori's.

V. Research Methodology

Database has been used in business management, government administration, scientific and engineering data management and many other important applications. The recently extricated data or learning might be connected to data administration, query processing, process control, decision making and numerous other helpful applications. With the explosive development of data, mining data and information from huge databases has turned out to be one of the significant difficulties for data administration and mining group. The successive itemset mining is persuaded by issues, for example, advertise wicker bin investigation [3]. A tuple in a market wicker bin database is an arrangement of things acquired by client in an exchange. An association rule mined from advertise wicker container database expresses that if a few things are bought in exchange, at that point it is likely that some different things are obtained also. Discovering every such rule is important for directing future deals advancements and store format. The issue of mining successive itemsets are basically, to find all rules, from the given value-based database D that have bolster more prominent than or equivalent to the client indicated least help. The present status and the pertinent work in the region of data mining by and large and in the range of association rules specifically; break down these works in the zone of mining successive itemsets; propose the new scheme for extracting the frequent itemsets based on hybrid approach of maximal Apriori and FPtree algorithm that has high efficiency in term of the time and the space; validate its efficiency and seek avenues for further research.

VI. Conclusion

The rapid development in computer technology made it possible to collect, store huge amount of data and apply data mining. Data mining aims to discover knowledge or patterns from the data especially large databases. There may be such situations that private data may be under violation because of gained knowledge or extracted knowledge by itself contains some private knowledge. Privacy preserving data mining arise from the need for do data mining without violating privacy of data or knowledge. Data hiding techniques preserve the private data while rule hiding techniques preserve the private rules or patterns. The aim of this research is to propose algorithms for privacy preserving frequent pattern mining. Since the characteristics of data repositories of different domains vary, each algorithm is analyzed using two different synthetic databases consisting of different characteristics, i.e., one database has long patterns with a low diversity of items and the other database has short patterns with a high diversity of items.

REFERENCES

[1] D. Agrawal and C. C. Aggarwal. On the design and quantification of privacy preserving data mining algorithms. In Proceedings of the 20th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, pages 247–255, Santa Barbara, California, USA, May 2001. ACM Press.

[2] M. Atallah, E. Bertino, A. Elmagarmid, M. Ibrahim, and V. Verykios. Disclosure limitation of sensitive rules. In Proceedings of 1999 IEEE Knowledge and Data Engineering Exchange Workshop (KDEX'99), pages 45–52, Chicago, Illinois, USA, November 1999. IEEE Computer Society.

[3] M. Atzori, F. Bonchi, F. Giannotti, and D. Pedreschi. Blocking anonymity threats raised by frequent itemset mining. In Proceedings of the 5th IEEE International Conference on Data Mining (ICDM'05), pages 561–564, Houston, Texa, USA, November 2005. IEEE Computer Society.

[4] R. Agrawal, A. Evfimievski, and R. Srikant. Information sharing across private databases. In Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data, pages 86–97, San Diego, California, USA, June 2003. ACM Press.

[5] J. Alber, M.R. Fellows, and R. Niedermeier. Efficient data reduction for dominating set: A linear problem kernel for the planar case. In M. Penttonen and E. M. Schmidt, editors, Proceedings of the 8th Scandinavian Workshop on Algorithm Theory (SWAT'02), pages 150–159, Turku, Finland, July 2002. Springer Verlag Lecture Notes in Computer Science.

[6] A. Kaur, V. Aggarwal and S. K. Shankar, "An efficient algorithm for generating association rules by using constrained itemsets mining," *2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, Bangalore, 2016, pp. 99-102.

[7] B. M. Al-Maqaleh and S. K. Shaab, "An Efficient Algorithm for Mining Association Rules Using Confident Frequent Itemsets," *2013 Third International Conference on Advanced Computing and Communication Technologies (ACCT)*, Rohtak, 2013, pp. 90-94.

[8] P. Dong and B. Chen, "New Algorithm of Maximum Frequent Itemsets Based on FP-Tree for Mining Multiple-Level Association Rules," *2008 International Conference on Computer Science and Software Engineering*, Wuhan, Hubei, 2008, pp. 263-266.

[9] P. Dong and B. Chen, "New Algorithm of Maximum Frequent Itemsets for Mining Multiple-Level Association Rules," *2008 3rd International Conference on Innovative Computing Information and Control*, Dalian, Liaoning, 2008, pp. 332-332.

[10] Pei-Qi Liu, Zeng-Zhi Li and Yin-Liang Zhao, "Effective algorithm of mining frequent itemsets for association rules," *Proceedings of 2004 International Conference on Machine Learning and Cybernetics (IEEE Cat. No.04EX826)*, 2004, pp. 1447-1451 vol.3.

[11] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In Proceedings of the ACM SIGMOD Conference on Management of Data, pages 207–216, New York, NY, USA, May 1993. ACM Press.

[12] A. Ambainis, M. Jakobsson, and H. Lipmaa. Cryptographic randomized response techniques. In F. Bao, R. H. Deng, and J. Zhou, editors, Proceedings of the 7th International Workshop on Theory and Practice in Public Key Cryptography,, volume 2947, pages 425–438, Singapore, March 2004. Springer Veralg Lecture Notes in Computer Science.

[13] G. Apostolopoulos, V. G. J. Peris, and D. Saha. Transport layer security: How much does it really cost? In Proceedings of the Conference on Computer Communications (INFOCOM'99), joint conference of the (IEEE) Computer and Communications Societies, pages 717–725. IEEE Computer Society, March 1999.

[14] R. Agrawal and R. Sriknat. Fast algorithms for mining association rules in large databases. In Proceedings of the 20th International Conference on Very Large Databases (VLDB'94), pages 487–499, Santiago, Chile, September 1994. Morgan Kaufmann Publishers Inc.

[15] R. Agrawal and R. Srikant. Privacy-preserving data mining. In Proceedings of the ACM SIGMOD Conference on Management of Data, pages 439–450. ACM Press, May 2000.

[16] J. Benaloh. Dense probabilistic encryption. In Proceedings of the Workshop on Selected Areas of Cryptography, pages 120–128, Kingston, Ontario, Canada, May 1994.

[17] I. Bhandari, M. J. Halliday, J. Chaar, R. Chillarence, K. Jones, J. S. Atkinson, C. Lepori-Costello, P. Y. Jasper, E. D. Tarver, C. C. Lewis, and M. Yonezawa. In-process improvement through defect data interpretation. In IBM Systems Journal, volume 33(1), pages 182–214, Riverton, NJ, USA, 1994. IBM Corp.

[18] J. E. Bartlett, J. W. Kotrlik, and C Higgins. Organizational research: Determining appropriate sample size for survey research. In B. N. OConnor, editor, Information Technology Learning, and Performance Journal, volume 19(1), pages 43–50, Spring 2001.