Basic Pattern Mining: A Review on Techniques and Applications

Bibi Noor Asmat, Muheb Ullah, Dr. Rukhshanda

Abstract --- With the rapid growth in technology, managing this data is becoming a challenging task. Data mining is the field of automatic extraction of information from a large corpus of data. Pattern mining is a technique initially used for market basket analysis, in order to extract the combination of items a customer buys at once. Frequent pattern mining algorithms can be categorized in two major types: Basic and extended pattern mining algorithms. This paper is basically concerned with basic type frequent pattern mining algorithms and presents a broad overview of these algorithms. These algorithms have been studied in terms of methodology they used, the focus of the algorithm and the purpose of these algorithms.

Index Terms: Data Mining, Frequent pattern mining, association rules.

1. INTRODUCTION

WE are drowning in information but starved for knowledge.

Famous quote of John Naisbitt becomes true when we are dealing with knowledge discovery from today's world of data. With the rapid growth of technology, the amount of data is also growing as an astonishing rate. Knowledge discovery from such a large amount of data is one of the hot challenges of today's world of data. Data mining is a solution to resolve this issue of interest. Data mining is the automatic extraction of patterns, sub-sequences and sub-patterns from huge corpus of data. In other words, the process of revealing hidden and useful information from a huge set of data is known as data mining. (Yen et al, 2009)

Pattern mining is a data mining technique used to extract different patterns or sequences from data automatically. Pattern mining is based on objects frequencies and the input is given as transactional form. Items, Terms and objects are interchangeably used in literature to represent a single entity or object of the data, but the word "item" is commonly used by researchers. Pattern mining has different applications depending on the nature of the extracted pattern. Transaction is an instance of items that occur collectively, e.g. set of items that a specific customer buys at a specific time in a market. A transaction may be a click stream, a combination of objects detected by a video surveillance system at a specific time interval or calls made by a specific user at a given time interval. Basic and extended Patterns mining algorithms are commonly used in literature and a variety of algorithms are proposed by researchers.

Basic patterns algorithms were initially proposed by the pioneers of pattern mining era. These are the basic techniques and have high familiarity as compared to extended patterns. Following are the main categories of basic pattern mining algorithms: Frequent Itemset Mining, Association Rule Mining and Frequent Closed/ Maximal Pattern Mining.

Extended patterns are based on basic patterns and variations of the earlier ones, enabling them to find any other sort of information as per requirement. Extended patterns normally have the same concepts as basic ones but have variations of conditions or constraints. Extended patterns includes Top-k Item-set mining, Uncertain data mining, Frequent periodic patterns, High utility patterns mining, Rare Itemset Mining and Colossal patterns mining.

As discussed earlier, this paper is basically concerned with basic pattern mining algorithms. Following are the brief discussion of the major types of basic pattern mining algorithms and techniques:

2. FREQUENT ITEMSET MINING

Frequent Patterns are subsets, subsequence, sub patterns or Itemset that occur collectively at a frequency higher than a threshold value. This threshold is known as support value or min-sup and is user specified. The support value is the ratio of the transactions supporting a particular item or object to the total number of transactions present in a dataset. If a user selects a higher support value say 90% support, the items that lie in 90% of the transactions will be returned and thus only most frequent items will be returned. On the other side if a user selects 10% support, algorithm will returns the items that lie in 10% transactions only. Thus higher support value minimizes the possibilities and thus returns less number of Frequent Itemset (FI) and a smaller support value returns larger number of FIs.

Frequent Patterns, frequent Itemset and frequent term sets are interchangeably used in literature. Frequent patterns are a general term used to represent this concept. Word frequent Itemset is used when the transactions are individual objects e.g. food items in case of market analysis and frequent term sets are particularly used in the area of text mining. R. Agrawal, T. Imielinski, A. Swami [2] and R. Agrawal, R. Srikant [3] introduced the concept of frequent patterns or FIs.

"Given the min-sup threshold, an Itemset I is called frequent Itemset when supp(I) is greater than or equal to min-sup, i.e., $FI = \{X|Supp(I) \geq min\_sup$, where $I = i1, i2 , i3, ...im$" [17].

Frequent pattern mining for the very first time was applied to the market basket analysis but now it's applications are becoming vast day by day. Some suitable applications for FIs are analysis of market, text mining, clustering, click-streams, and network traffic.

2.1 ALGORITHIMS FOR MINING FREQUENT ITEMSET

Researchers have proposed different algorithms and techniques for extraction of FIs that improve efficiency, scalability and preserve memory as well. Scalability means the ability of the algorithm when the size of input data is increased. Following is a brief discussion of FI extraction algorithms for situations and conditions:

TABLE I
LITERATURE SUMMARY OF FREQUENT ITEM-SET MINING ALGORITHMS.

S.No	Author and year	Algorithm	Description	Focus
1	(Agrawal & Srikant, 1994)	Apriori	Based on candidate set generation, term frequencies. Increment size of candidate set in ascending order.	Finding FIs in market data analysis.
2	Zaki, 2000	Eclat	Organize the items using lattice search space and sub-lattice, small independent memory chunks, solvable in memory. To identify all long patterns and sub-patterns, they presented efficient traversal techniques.	Efficiency
3	(Zaki & Gouda, 2001, 2003)	dEclat	Vertical mining approach. Uses diffset, a compact form of data representation.	Scalability
4	Uno et al., 2004	LCMFReq	Fast enumeration based on Prefix preserving closure (PPC) extension, an extension from one closed item-set to another. Uses search tree for all	Efficiency and preserves memory.

			closed item-set without regenerating the extracted item-sets again.	
5	(Han et al., 2004)	FP-Growth	Frequent Pattern Tree, used divide and conquer method. Frequent pattern tree is a compact form of hierarchical representation of patterns that preserves memory and is efficient to scan.	Reducing repeated database scans, and number of candidate set.
6	(Borgelt, 2005)	Relim	Inspired from FP-Growth algorithm but does not use pre-fix tree or any other complex data structure. Recursive elimination algorithm.	Simplicity of the structure
7	(Pei et al., 2007)	H-Mine	Based on data structure H-Struct, and hyperlinks. H-struct requires limited memory and is efficient to run in memory. Divides database in small partitions. Proposed a space preserving mining algorithm as well.	Scalability
8	Deng et al., 2012, Deng et Lv, 2015	PrePost, PrePost+	Vertical data representation method, N-List is used. Originated from FP-Tree and PPC-Tree. Compactness of N-List and use of intersection of two N-lists.	Efficiency but more memory consumption for sparse datasets.
9	(Deng et al., 2014)	FIN	Inspired from PrePost algorithm implemented using PPC-Tree, based on Nodeset data structure. Node-Set is more efficient data structure than N-List and Node-List used in PrePost.	Efficiency and memory preserving

From the above table we can see that that initially FIs algorithms used candidate sets as used in Apriori. With the arrival of Eclat algorithm, researcher started following hierarchical a approach that is more compact form of pattern representation. Vertical data mining technique was used by dEclat algorithm for the first to increase efficiency. FP-Growth for the fisrt time used proper tree structure, FP-Tree a more compact form of tree as compare to Eclat algorithm, and thus improved efficiency and memory consumption. Relim was inspired from FP-Growth and was simpler but it prefers simplicity over efficiency and the algorithm is inefficient in some cases. Rest of the algorithms like H-Mine, PrePost and FIN used different efficient data structures like H-Struct, N-List and PPC-Tree inorder to improve efficiency and they added different efficient traversal techniques as well.

2.2 ALGORITHMS FOR MINING FREQUENT ITEMSET USING MULTIPLE MINIMUM SUPPORTS

As discussed earlier, frequent Itemset are extracted using min-sup, a user specified threshold value. A very high min-sup leads to the extraction of very few item-sets, because the item-sets with low support are discarded. Setting min-sup very low, returns to a large number of frequent Itemset. In order to preserve memory, a user try to set min-sup high, but on the other hand high min-sup will not return all the frequent Itemset required. This will sometimes leads to the loss of information unwillingly. To cope with this problem, different researchers have proposed variety of algorithms that uses multiple min-sup. Following is the summary of these algorithms, the methodologies and their technical contribution in terms of memory, run time and scalability etc:

From the below table, it is concluded that algorithms to use multiple min-sup are basically variations of the well-known algorithms Apriori and FP-Growth algorithms. This is one of the open areas for researchers that can be addressed in future.

TABLE II
LITERATURE SUMMARY OF FREQUENT ITEM-SET MINING ALGORITHMS WITH MULTIPLE SUPPORT

S.No	Author and year	Algorithm	Description	Focus
1	(Liu et al, 1999)	MSApriori	Based on Apriori algorithm and uses multiple supports to reveal the nature of Itemset in the database and diverse frequencies. Minimum Item Supports (MIS) is used to express the min-sup of the items present in a rule.	Multiple Min-Support
2	Hu & Chen, 2006, Uday & Reddy, 2011,	CFP-growth CFPGrowth++	Based on FP-Growth algorithm but accepts multiple supports, uses MIS-Tree, an FP-Tree like structure. Each item has its own min-sup, so it is difficult to select proper min-sup. CFPGrowth algorithm works repeatedly until a satisfactory result is fond. Efficient algorithm for scanning MIS-Tree is proposed to make the process efficient. CFPGrowth++ is an extension of CFPGrowth with addition of new pruning techniques.	Efficient and Scalable

2.3 ALGORITHMS FOR MINING FREQUENT ITEMSET FROM UNCERTAIN DATA

Managing uncertain and imprecise data is one of the hot issues in the area of pattern mining. Example of uncertain data includes collecting data about temperature in a habitat monitoring system where the sensor normally produces noisy data. Another example is finding location of a person using GPS system that produces imprecise data. For handling such type of data, probabilistic frameworks are implemented. Data that is not certain is handled using probabilistic models.

Uncertain transactions have associated probabilities and works with probabilistic framework. Using conventional algorithms for extraction of frequent Itemset from uncertain data is appropriate approach. Few algorithms to deal with such type of data is given in table 3.

U-Apriori simply finds frequent Itemset while P-Apriori computes Top-A frequent Itemset. Both the algorithms are based on probabilistic framework.

TABLE III
LITERTERATURE SUMMARY OF FREQUENT ITEM-SET MINING ALGORITHMS FROM UNCERTAIN DATA.

S.No	Author and year	Algorithm	Description	Focus
1	(Chui et al, 2007)	U-Apriori	It uses uncertain data under probabilistic framework. Items of the transactions have associated probabilities and give a formal definition of frequent patterns in such uncertain environment.	Saves CPU and I/O cost
2	Feng Gao et al, 2011	P-Apriori	It is an algorithm based on probabilistic model, using dynamic programming for testing frequent item-sets. It mines top-A probabilistic frequent Itemset, and reports Itemset incrementally in ascending order.	Efficiency and effectiveness

2.4 ALGORITHMS TO DISCOVER FREQUENT ITEMSET FROM A STREAM

Stream of data is an unbounded sequence of data, generating continuously at rapid rate. This continuing generation of data changes the result with passage of time as the new data is generated from the stream. New addition of data reduces the effect of old data and thus making the task more complicated. To deal with such type of data, researchers have presented some algorithms that specifically deal with transactions extracted from data streams. (Chang & Lee, 2003) Following is a summary of some well known algorithms used for extraction of frequent patterns generated from data streams:

TABLE IV
LITERATURE SUMMARY OF FREQUENT ITEM-SET MINING ALGORITHMS FROM DATA STREAM.

S.No	Author and year	Algorithm	Description	Focus
1	(Chang & Lee, 2003)	estDec	Adaptively computes frequent item-set from recently obtained data stream. The effect of old item-sets diminishes with the time as new data-streams are received. Optimization techniques are proposed.	Efficient and minimizing memory usage
2	(Yen et al, 2009)	CloStream	Computes frequent closed item-set from a data stream. Incrementally updates the item-sets as per user specified threshold.	Efficient and minimizing memory usage
3	(Shin et al., 2014)	estDec+	Uses Compressible-Prefix Tree (CPT) that can trace the support of multiple items at once. Further the size of CPT decreases as the number of traced items increases. CPT consumes less memory but produces less accurate Frequent item-sets. To control the size of CPT, estDec+ uses a splitting and merging of nodes of the tree. A threshold value for merging gap is also supported which controls the utilization of confined memory.	Efficient and minimizing memory usage

estDec simply computes Fis as new data are generated from the stream. CloStream works in same manner but it returns frequent closed Itemset. A detailed description of frequent closed Itemset is given in the section of frequent closed Itemset algorithm. estDec+ is inspired from estDec but uses tree structure that is more compact and efficient.

3 ASSOCIATION RULE

Association rule mining is the process of discovering frequent patterns, correlations and associations between two or more than two items. Formerly, association rule is in the form $X \Rightarrow Y$, where $X \subseteq I$, $Y \subseteq I$, where $I = \{i1, i2, i3 \dots, im\}$ is a set of items. The rule $X \Rightarrow Y$ holds in the transaction set D with confidence C if C% of transactions in D that contain X also contain Y . The rule $X \Rightarrow Y$ has support s in the transaction set D if S% of transactions in D contain $X \cup Y$. [47]

Following are some well known algorithms for computing association rules:

TABLE V
LITERATURE SUMMARY OF ASSOCIATION RULE MINING ALGORITHMS.

S.No	Author and year	Algorithm	Description	Focus
1	(Agrawal & Srikant, 1994)	AprioriHybrid	It combines Apriori and AprioriTID for the purpose of efficiency and scalability. It uses Apriori initially and then switches to AprioriTID when it expects that a candidate at the end of the pass will fit in memory.	Efficiency and scalability
2	(Koh & Roundtree, 2005)	Apriori-Inverse	Use to mine sporadic rules, rules with low support and high confidence level. Divide rules in two categories. Perfectly sporadic rules are those rules having items with support below maximum support, and imperfect sporadic rules are those having items with support larger than maximum support.	Efficiency
3	(Tan et al. 2000; Tan et 2006)	INDIRECT	Introduces a novel pattern called indirect association. Patterns that posses support higher than user specified threshold. Infrequent pair of items can be useful if the items are related indirectly via some other set of items. INDIRECT algorithm extracts indirectly associated item pairs.	Estimate the correlation between indirect data
4	(Weng et al. 2008)	FHSAR	Privacy-preserving algorithm. Sensitive data are given high disclosure threshold and is kept hidden from the network or user. Information is represented in the form of frequent item-set or association rule.	Efficiency
5	(Fournier-Viger, 2012b)	TopKRules	Normally algorithms generate large amount of FI's. TopKRules returns only k association rules with highest support value.	Scalability, user control over number of FIs.
6	(Fournier-Viger 2012)	TNR	Extracts top-k non redundant association rules.	Scalability and accuracy.

From the above table it is concluded that algorithms for association rule mining does not only improve efficiency but they have their own specific purpose as well. AprioriHybrid uses association rule and were followed by many researchers as their base algorithm. AprioriInverse discovers sporadic rules, INDIRECT and FHSAR have their own specific purpose along with the extraction of association rules. TopKRules and TNR both returns top k association rules. TNR algorithm computes non-redundant top-k association rules as discussed above.

4. CLOSED/MAXIMAL FREQUENT ITEMSET MINING ALGORITHMS

Existing FI extracting techniques are not effective as they produce a very huge amount of FIs where most of these FIs are not useful. To extract different interesting and useful FIs, many alternatives have been presented by researchers including mining closed patterns, maximal patterns and maximum length itemset. Currently most of the algorithms use monotonicity property of the FIs which states that if an FI is frequent then all its sub-sets are also frequent. These approaches are discussed as following:

Frequent Closed Itemset (FCI): "*A pattern a is a closed frequent pattern in a data set D if a is frequent in D and there exists no proper superset β such that β has the same support as a in D*" (Han et al., 2007). [18]

Maximal Frequent Itemset (MFI): "*A pattern a is a maximal frequent pattern (or max-pattern) in set D if a is frequent, and there exists no superset β such that a sub-set β and β is frequent in D*" (Han et al., 2007). [18]

Following is a brief summary of the some well-known algorithms for extracting Frequent Closed item-sets and Maximal Frequent item-sets:

TABLE VI
LITERATURE SUMMARY OF CLOSED/MAXIMAL FREQUENT ITEM-SET MINING ALGORITHMS.

S.No	Author and year	Algorithm	Description	Focus
1	(Pasquier et al., 1999)	AprioriClose	Also known as A-Close algorithm, reducing sub-set latice to closed lattice for reducing search space.	Efficient for dense and correlated data.
2	Doug et al, 2000	Mafia	It uses search strategy using combination of depth-first traversal of the Itemset lattice with effective pruning mechanisms.	Efficiency
3	Jian Pei et al, 2000	CLOSET +	Using a compressed tree structure, FP-Tree. Explores closed frequent item-sets.	Efficiency and scalability
4	o (Zaki and Gouda, 2001)	dCharm	Vertical data mining using diffset data set. Inspired of Charm algorithm. Advantage of having support for fast frequency counting via intersection operations on transaction ids (tids) and automatic pruning of irrelevant data. Extracts closed and maximal Itemset.	Preserves memory and improves efficiency.
5	(Zaki and Hsiao, 2002)	Charm	Uses dual Itemset tid-set search tree with combination of hybrid search technique that excludes many levels in tree traversal. For memory preserving it uses diffset, an already discussed data structure. A hash based approach for removing non-closed patterns.	Scalability
6	(Grahne and Zhu, 2003)	FPMax	Extension of FP-Growth algorithm. Extracts maximal frequent Itemset.	Focused on efficiency but claimed that efficiency depends on tuning of parameters, different results for different environments.

7	(Uno et al., 2004)	LCMmax	Inspired of LCM as discussed above, explores maximal and closed Itemset using LCM algorithm.	Efficiency
8	(Lucchese et al, 2004)	DCI_Closed	Vertical data mining using depth-first closed item-set mining algorithm. Detects duplicate item-sets and explores the non-redundant closed item-sets.	Preserves memory.
9	Grahne, G., & Zhu, J. (2005)	FPCLOSE	FP-array based technique for extracting Frequent closed item-sets. Efficient traversal techniques are used to improve run time.	Efficient and preserve memory for dense datasets.
10	Karam et al, 2005	Genmax	It uses a novel technique called progressive focusing to perform maximality checking, and diffset propagation to perform fast frequency computation.	Efficiency

FPMax and LCMmax both are based on their parent algorithms FP-Growth and LCM but returns closed/maximal patterns. FPCLOSE and Genmax uses FP-Array and diffset to increase efficiency and scalability. FP-Array's compactness helps FPCLOSE to preserves memory as well.

5. CONCLUDING REMARKS

In this paper we conclude that data and transactions may of different nature, only single algorithm cannot be used to extract frequent patterns. Nature of data is a most important factor that cannot be neglected. Transactional data, data streams, colossal, monotonic and episodes data all have their own specific algorithms. To achieve efficiency, researchers have used tree structures and efficient traversal techniques and new data types are proposed. In future work, a brief literature study on extended patterns is highly recommended. Furthermore a comparative study of different algorithms can be carried out in terms of efficiency, memory and other factors used above. The algorithms can be experimented using different data structures that are used above.

REFERENCES:

[1]. U. Fayyad, G. Piatetsky-Shapiro, P. Smyth (1996) "From Data Mining to Knowledge Discovery: An Overview", *Journal of record for the AI community: AI Magazine*, **vol. 17,** MIT Press, Cambridge, Mass, pages. 37-48.

[2]. R. Agrawal, T. Imielinski, A. Swami (1993) "Mining association rules between sets of items in large databases" Proceedings of the 1993 ACM SIGMOD international conference on "**Management of data**", Washington DC, USA, MIT Press, New York, pages. 207-216.

[3]. R. Agrawal, R. Srikant (1994) "Fast algorithms for mining association rules in large

databases”, Proceedings of the 20th International Conference on **“Very Large Data Bases, VLDB”**, Santiago, Chile, pages. 487-499.

[4]. R.Jensi, Dr.G.Wiselin Jiji (2013) “A Survey On Optimization Approaches To Text Document Clustering”, *International Journal on Computational Sciences & Applications (IJCSA)*, **vol.3**, pages. 31-44.

[5]. J. A. Hartigan, M. A. Wong (1979) “Algorithm AS 136: A K-Means Clustering Algorithm”, *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, **vol. 28,** pages. 100-108.

[6]. P. S. Bradley, O. L. Mangasarian, W. N. Street (1997) “Clustering via Concave Minimization”, *Journal of Advances in Neural Information Processing Systems,* **vol. 9**, pages.368-374.

[7]. R. Ng, J. Han (1994) “Efficient and effective clustering method for spatial data mining”, Proceeding of the 20th Conference on **“VLDB”**, Santiago, Chile, pages. 144–155.

[8]. N. M. AbdelHamid, M. B. A. Halim, M. W Fakhr (2013) “Bees Algorithm-Based Document Clustering”, The 6th International Conference on **“Information Technology”**, College of Computing and Information Technology, Cairo, Egypt, pages. 246-254.

[9]. K. Sasirekha, P. Baby (2013) “Agglomerative Hierarchical Clustering Algorithm- A Review Similarity Measures”, *International Journal of Scientific and Research Publications*, **vol. 3,** pages. 1515-1518.

[10]. S. Guha, R. Rastogi, K. Shim (1999) "ROCK: A Robust Clustering Algorithm for Categorical Attributes," Proceeding of the 15th International conference on **“Data Engineering”,** IEEE CS Press, Los Alamitos, Calif, pages. 512-521.

[11]. G. Karypis, E. Han, V. Kumar (1999) "Chameleon: Hierarchical Clustering Using Dynamic Modeling", *IEEE Computer Special Issue on Data Analysis and Mining*, **vol.32,** pages. 68-75.

[12]. F. Beil, M. Ester, X. Xu (2002) “Frequent Term-Based Text Clustering”, Proceedings of the eighth ACM SIGKDD international conference on **“Knowledge discovery and data mining”**, ACM, New York , pages. 436-443

[13]. B. Fung, K. Wang, M. Ester (2003) “Hierarchical Document Clustering Using Frequent Itemsets”, Proceeding of SIAM International Conference on **“Data Mining”**, SIAM, pages.59-70.

[14]. C. Su, Q. Chen, X. Wang, X. Meng (2009) "Text Clustering Approach Based On Maximal Frequent Term Sets", Proceeding of 2003 IEEE International Conference on **“Systems, Man and Cybernetics**", Harbin Institute of Technology, Shenzhen, China, pages.1551-1556.

[15]. G. Salton, A. Wong, C. S. Yang (1975) “A Vector Space Model for Automatic Indexing”, *Journal of Communications of the ACM*, **vol.18,** pages.613-620.

[16]. D. Burdick,M. Calimlim, J. Gehrke (2001) "MAFIA: a maximal frequent itemset algorithm for transactional databases" Proceedings of 17thInternational Conference on **“Data Engineering”**, pages.443-452.

[17]. A. Salam (2011) "Mining Frequent Patterns Without Minimum Support Threshold", International Islamic University, Islamabad, PhD thesis.

[18]. J. Han, H. Cheng, D. Xin, X. Yan (2007) "Frequent Pattern Mining: Current Status and Future Directions", *Journal of Data Mining and Knowledge Discovery,* **vol. 15**, pages. 55-86.

[19]. MJ. Zaki, CJ. Hsiao (2002) "CHARM: An efficient algorithm for closed itemset mining", Proceedings of the 2nd SIAM international conference on **"Data Mining",** IEEE Educational Activities Department, Piscataway, NJ, USA, pages. 12–28

[20]. T. Hu, X. Wang, Q. Fu, S. Y. Sung (2006) "Mining Maximum Length Frequent Itemsets: A Summary of Results"18th IEEE International Conference on **"Tools with Artificial Intelligence",** ICTAI '06, Arlington, VA, pages.505-512.

[21]. N. Pasquier, Y. Bastide, R. Taouil, L. Lakhal (1999) "Discovering Frequent Closed Itemsets for Association Rules" Proceedings of the 7th International Conference on **"Database Theory",** ICDT '99, Springer-Verlag, London, UK, pages. 398-416.

[22]. J. Pei, J. Han, R. Mao (2000) "CLOSET: An Efficient Algorithm for Mining Frequent Closed Itemsets" ACM SIGMOD Workshop on **"Research Issues in Data Mining and Knowledge Discovery",** pages. 21-30.

[23]. J. Wang, J. Han, J. Pei. (2003) "CLOSET+: searching for the best strategies for mining frequent closed itemsets" Proceedings of the ninth ACM SIGKDD international conference on **"Knowledge discovery and data mining",** (KDD '03), ACM, New York, NY, USA, page. 236-245.

[24]. RJ. Bayardo (1998) "Efficiently Mining Long Patterns from Databases". Haas LM, Tiwary A, eds. Proceedings of ACM SIGMOD International Conference on **"Management of Data",** pages. 85-93.

[25]. L. Ertoz, M. Steinbach, V. Kumar, (2003) "Finding Clusters of Different Sizes, Shapes and Densities in Noisy, High Dimensional Data", Proceedings of the Third SIAM International Conference on **"Data Mining**"(SDM 2003)', Society for Industrial and Applied Mathematics.

[26]. A. H. Lashkari, F. Mahdavi, V. Ghomi (2009) "A Boolean Model in Information Retrieval for Search Engines "International Conference on **"Information Management and Engineering",** ICIME '09, IEEE Computer Society, pages. 385-389.

[27]. M.S. Chen, J. Han, P.S. Yu (1996) "Data Mining: An Overview from a Database Perspective", *IEEE Transactions on Knowledge and Data Engineering*, **Vol. 8,** pages. 866-883.

[28]. C Jin, MMA Patwary, A Agrawal, W Hendrix, W Liao, A. Choudhary (2013) "DiSC: A Distributed Single-Linkage Hierarchical Clustering Algorithm using Map Reduce" Proceedings of the 4th International SC Workshop on "**Data Intensive Computing in the Clouds**", pages. 555-561

[29]. J. A. Hartigan, M. A. Wong (1979) "Algorithm AS 136: A K-Means Clustering Algorithm", *Journal of the Royal Statistical Society*, Series C , **vol. 28**, pages. 100–108.

[30]. T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, A. Y. Wu, (2002) "An efficient k-means clustering algorithm: Analysis and implementation", *Journal of IEEE Trans on Pattern Analysis and Machine Intelligence*, **vol. 24**, pages. 881–892.

[31]. C. Ding, X. He (2004) "K-means Clustering via Principal Component Analysis", Proceeding of *the twenty-first* International Conference on **“Machine Learning”**(ICML 2004), ACM, pages. 225–232.

[32]. S. Har-Peled, S. Mazumdar (2004) “On core sets for k-means and k-median clustering”, Proceedings of the 36thannual ACM symposium on **“Theory of computing”,** (STOC '04), ACM, pages. 291-300.

[33]. K. Murugesan, Z. Changjiang (2011) "Hybrid Bisect K-Means Clustering Algorithm," International Conference on **“Business Computing and Global Informatization (BCGIN)”,** pages. 216-219.

[34]. Y. Li, S. M. Chung (2007) “Parallel bisecting k-means with prediction clustering algorithm”, *Journal of Supercomputing*, **vol.** 39, pages.19-37.

[35]. M. Steinbach, G. Karypis, V. Kumar (2000)“A Comparison of Document Clustering Techniques” KDD Workshop on “Text Mining” University of Minnesota, pages. 00-34.

[36]. Ng, R.T, J. Han (2002) "CLARANS: a method for clustering objects for spatial data mining", *Journal of IEEE Transactions on Knowledge and Data Engineering*, **vol.14,** no.5, pages.1003-1016.

[37]. J. Han, J. Pei, Y. Yin (2000) “Mining frequent patterns without candidate generation”, Proceedings of the 2000 ACM SIGMOD international conference on **“Management of data”**, Dallas, Texas, USA, pages. 1-12.

[38]. J. Han,J. Pei, Y. Yin, R. Mao (2004) “Mining Frequent Patterns Without Candidate Generation: A Frequent-Pattern Tree Approach”, *Journal of Data Min. Knowl. Discov*, **vol. 8**, issue. 1, pages. 53 - 87.

[39]. C. Gyorödi, R. Gyorödi, S. Holban (2004) "A Comparative Study of Association Rules Mining Algorithms", SACI 2004, 1st Romanian-Hungarian Joint Symposium on **“Applied Computational Intelligence”** , Timisoara, Romania, pages. 213-222.

[40]. V. TUNALI(18 Nov, 2014) “Classic3 and Classic4 DataSets”, http://www.dataminingresearch.com/index.php/2010/09/classic3-classic4-datasets.

[41]. LABIC- Laboratory of Computational Intelligence, (18 Nov, 2014) “Index of /torch/datasets”, http://sites.labic.icmc.usp.br/torch/datasets/.

[42].S. Christa, V. Suma, L. Maduri (2012) “An Effective Data Preprocessing Technique for Improved Data Management in a Distributed Environment”, *International Journal of Computer Applications*, IJCA, Special Issue on “Advanced Computing and Communication Technologies for HPC Applications**,** ACCTHPCA”, **vol.3**, pages. 25-29.

[43]. R. Sedgewick, K. Wayne (18th Nov 2014) “Sop Word List” http://algs4.cs.princeton.edu/35applications/stopwords.txt

[44]. (18thNov 2014) “e-lemma list” http://stel.ub.edu/sites/default/files/e_lemma.txt

[45]. M. Rosell, V. Kann, J. E. Litton (2004)“Comparing comparisons: Document clustering evaluation using two manual classifications” International Conference on **“Natural Language Processing”,** Allied Publishers Private Limited, pages. 207-216.

[46]. S. Banerjee, K. Ramanathan, A. Gupta (2007)"Clustering short texts using wikipedia" Proceedings of the 30th annual international ACM SIGIR conference on **"Research and development in information retrieval**, (SIGIR '07)", ACM, pages. 787-788.

[47]. Rakesh Agrawal and Ramakrishnan Srikant. 1994. Fast Algorithms for Mining Association Rules in Large Databases. In *Proceedings of the 20th International Conference on Very Large Data Bases* (VLDB '94), Jorge B. Bocca, Matthias Jarke, and Carlo Zaniolo (Eds.). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 487-499.