E FAR-HD: ENHANCED FUZZY ASSOCIATION RULE MINING USING JARO-WINKLER DISTANCE FOR HIGH DIMENSIONAL DATASETS

Dhiraj Kapila
Assistant. Professor (C.S.E)
D.A.V Institute of Engg & Technology
Jalandhar, Punjab, India

Ms. Harpreet Bajaj
Associate . Professor ( C.S.E )
D.A.V Institute of Engg & Technology
Jalandhar, Punjab, India

Abstract— Association Rule Mining (ARM) with fuzzy logic theory accelerates the easy process of mining of latent frequent or recurrent patterns constructed on their own frequencies in the form of association rules from any transactional and relational datasets containing objects and items to imply the most recent trends in the given dataset. These mined recurrent patterns or fuzzy association rules employ either for physical data analysis or also influenced to compel further mining tasks like categorization (classification) and collecting (clustering) which helps domain area experts to systematize decision-making. In the conception of data mining, generally fuzzy Association Rule Mining (FARM) technique has been expansively adopted in transactional and relational datasets those datasets containing objects and items who have a fewer to medium amount of attributes/dimensions. Few techniques have also implemented for high dimensional dataset also, but whether those techniques have also works well for low dimensional datasets are yet to be proven out. Hence, in this paper we propose E-FAR-HD algorithm which is an enhanced version of FAR-HD algorithm that premeditated absolutely for large or high-dimensional datasets. We have intended this EFAR-HD algorithm that increases the accuracy of FAR-HD algorithm on the smaller datasets and remove the chances of misses when FAR-HD has tested on smaller datasets such as contact lens or patient dataset.

Index Terms—Fuzzy Association Rule Mining, Fuzzy Clustering, Fuzzy Partitioning, Fuzzy Relations, Partitions, Tidlists, High Dimensions, Large Datasets, Smaller Datasets,

1. INTRODUCTION

Data mining is the procedure to dig out the inherent information and knowledge from the collection of, inadequate, imperfect, noisy, fuzzy, random and disorganized data which is hypothetically functional and people do not know in advance about this concealed and hidden information [65]. The imperative difference between the outmoded data analysis technique such as query reporting and the data mining and is that the data mining is very supportive and extremely helpful to conclude knowledge and also useful in mining information based on the proposition of no clear hypothesis [66]. The most essential use of data mining is in programmed data analysis technique to come across or to catch out previously unseen or
undiscovered associations among various data items in
the datasets.
Data mining is the comprehensive analysis step of the
"Knowledge Discovery in Databases" process, or
KDD),[45] which is an inter-corrective subarea of
computer science,[70][65][71] which is nothing but a
computational activity comprising of discovering
significant and hidden patterns and information in
large datasets containing objects and items. The
applications of data mining encompassing methods of
intersection of artificial intelligence, machine learning,
statistics, and database systems. [70] The overall
intention of the data mining procedure is to dig out
meaningful and hidden information from a dataset
containing items and then restore it into a rational
structure for forthcoming usage. [70] Apart from data
analysis step, it also encompasses the conceptions of
database and data management, data pre-processing.
Various other actions like inference and complexity
considerations, interestingness metrics, and post-
processing of discovered structures are also the part of
data mining process.
Association Rule Mining (ARM) is one of the most
domineering research area in the theory of data mining
that facilitate the mining of concealed recurrent
patterns that are based on their own frequencies in the
shape of association rules from any item set or datasets
containing objects and entities to characterize the
utmost recent trends in the given dataset. These mined
recurrent patterns or fuzzy association rules employ
moreover for physical data analysis or also prejudiced
to induce further mining tasks like categorization
(associative classification [25], [26], [27]) and
collecting (ARM- driven clustering, like document
clustering [28], [29], [20], [31]) which helps domain
area experts to systematize decision-making results.
Now a day’s fuzzy association rule mining has attain
incredible recognition because of its correctness or
accurateness, which can be ascribed to its capability to
mine large amounts of data from huge transactional
and relational datasets. Now frequent patterns retain
all the prevailing relationships between items and
entities in the given dataset and pact only with the
numerically noteworthy associations, classification or
clustering. Association rules mining technique in
widely used in various [40] areas such as
telecommunication networks, stock market research
and risk management, inventory control etc. The
Apriori algorithm is used for frequent item set mining
using association rules over the transactional
databases. The apriori algorithm is proceeds by
recognize the frequent individual items in the dataset
and expanding them to larger and larger item sets as
long as those item sets appear adequately often in the
database.
Association rule mining [64] is to find and dig out
association rules that gratify the pre-defined minimum
support and confidence from a given dataset of items.
In the concept of ARM, generally fuzzy Association
Rule Mining (FARM) technique has been
comprehensively adopted in transactional and
relational datasets those datasets containing items that
have a fewer to medium amount of
attributes/dimensions. Few techniques have also
adopted for high dimensional dataset also, but whether
those techniques have also work for low dimensional
datasets are yet to be proven out.
2. FAR-HD ALGORITHM
FAR-HD algorithm has been proposed and developed
by Ashish Mangalampalli and Vikram Pudi [4] which
is able to mine fuzzy association rules from high
dimensional datasets. As we know that the traditional
ARM algorithms like apriori and FP-growth look
forward for binary attributes and also these
conventional ARM algorithms cannot be applied directly on those datasets and in those fields, in which there is huge amount of contribution of numerical attributes or also have data with very large amount of numerical dimensions like picture datasets have.

The image domain dataset engages the feature vectors with more than 60 dimensions which require the efficient and resourceful algorithm that can able to mine fine association rules from or to carry out the operations like associative classification from this image dataset quickly. So FAR-HD algorithm is one of the good options to use to mine fuzzy association rules from these high-dimensional itemset. This is an efficient algorithm which can bale to mine fuzzy association rules from very high-dimensional numerical datasets that contain more than 0.5 million vectors and each vector length consists of at least 60 dimensions and corresponding to the outline of fuzzy features.

FAR-HD algorithm [36] employs fuzzy C-means (FCM) clustering method to generate fuzzy clusters from the feature vectors of the specified dataset. Each feature vector will be fit in to each of the k clusters with a definite level of membership which helps in dropping the problem of polysemy and synonymy which usually takes place in crisp clustering.

The prominent features of FAR-HD are that the algorithm embodies a two phased processing method, and a Tidlist scheme for manipulating the frequency of itemsets and also employs a nonspecific Zlib compression algorithm to compress Tidlist while handing out them in order to save many more Tidlist in the same amount of memory that is allocated or available to store them. Furthermore for, itemset generation and processing, this FAR-HD works well in DFS like manner in order to pact with those high dimensional datasets who have generated association rules with many items and their normal rule length is very high.

Further in this paper [4], Ashish Mangalampalli and Vikram Pudi has mentioned about the fuzzy pre-processing strategy and fuzzy actions that are employed for the authentic FAR-Mprocess. This preprocessing strategy executes in two steps. In this first step, there is a production of fuzzy clusters from the numerical vectors and during the second step there is a transformation of crisp dataset that contains numerical vectors into fuzzy datasets with the help of fuzzy-cluster-based representation. The objective of the algorithm in fuzzy pre-processing strategy is to minimize the following equation

\[ J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} \mu_{ij} m \left\| x_i - c_j \right\|^2 \] ..........................\( Eq. (1) \)

Where \( m \) is any real number in the range such that \( 1 \leq m < \infty \), and \( \mu_{ij} \) is the degree of membership of \( x_i \) in the cluster of \( j \), \( x_i \) is the \( i \)th dimensional measured data, \( c_{ji} \) is the \( d \)-dimensional cluster center, and \( \| * \| \) is any norm stating the comparison between any measured data and the center. The variable \( mm \) is known as fuzziness parameter which is an arbitrary real number \( (mm > 1) \). The quantity of fuzziness and Gaussian environment of fuzzy sets can be restricted via a suitable approximate value under the range of 1.1–1.5 of the fuzziness parameter \( mm \) (Eq. 1). Because of this reason, the consequent fuzzy separations of the dataset are produced where each assessment of numeric attributes are exclusively recognized by their membership functions (\( \mu \)). Based upon the amount of fuzzy separations have described for an attribute, each and every accessible crisp data is changed to compound fuzzy data. This conversion may lead to the leeway of combinatorial outburst of production of fuzzy records. So the authors have placed a low threshold value which is equal to 0.1 for the membership function \( \mu \) to control this numerous
production of fuzzy records. Throughout the FARM process, the novel crisp dataset is enlarged with element values within the range of (0, 1) because the huge amount of fuzzy separations are being done on every quantitative element. To execute this enlarged fuzzy dataset, a few procedures are required which are based on the term t-norms [43], [44], [45]. Due to this t-norm, the new fuzzy dataset E is created upon which the designed algorithm will work. The FAR-HD algorithm makes use of two phases in a division strategy to produce fuzzy association rules. The fuzzy dataset E is sensibly separated into pp disjoint flat separations P1, P2,..... Each separation is as big as it can fit in the accessible main memory. The authors have used the following notations,

- E = Fuzzy dataset based upon fuzzy-cluster-based representation produced after fuzzy pre-processing
- Sp = Set of singletons in existing separation pp
- td [it] = Tidlist of itemset iitt
- μp = collective fuzzy membership or fuzzy support of any itemset in existing partition pp
- [it] = collective μμ of itemset iitt over all separations p in which iitt has been executed
- d = number of partitions for some exacting itemset iitt that have been executed since the separation in which it was added.

FAR-HD algorithm structure employs a byte-vector-like data representation in which each cell accumulates μ of the itemset equivalent to the cell indicator of the tid to which the μ pertains.

Hence, the ith cell of the byte-vector includes the μ for the Ith tid. If an exacting transaction process does not enclose the itemset under concern, then the cell equivalent to that transaction process has allocated a 0 value. All the byte vectors in the cell have compressed using the well structured compression algorithm Zlib, prior before to be saved in the memory. In this way, they have gained a huge main memory space available at its disposal to speed up the execution and implementation of this FAR-HD [4] algorithm. As discussed earlier this FAR-HD algorithm employs two-phased approach, during the very early in the first phase the steps of FAR-HD Algorithm inspect each and every transaction in the existing partition of the item set, and generates a Tid list for each singleton originate. When all singletons in the existing partition have been listed or generated then the check is made to configure out which singleton is dd-frequent or not, the Tid lists having singletons who are seems not to be dd-frequent are dropped out. The formation of Tid list is carried out very soon as the new data set has been created. An item set is said to be dd-frequent if its incidence over dd partitions are equals to or surpasses the support accustomed for dd partitions, then the item set is considered to be frequent over dd partitions of the dataset E . Further the authors explained that the calculation of each and every singleton ss is preserved in the array data structure [ss]. To produce the bigger item sets, they the use depth-first search (DFS) traversing approach, i.e. starts with a singleton ssii and create all the supersets of ssii, prior to doing the similar for the subsequently singleton ssii+1. Primarily, every singleton ssii is united with one more singleton sj to create supersets of ssii in depth-first search manner. This progression is completed for each, sj where j = i + 1 to | Sp |. During the second phase, all the itemsets that has been appended in the existing partition in the first phase are also have been specified over the entire dataset E, and hence may be removed. From these removed data itemsets, those itemsets containing singletons which are d-frequent over the entire dataset EE are the output. This output dataset E is further logically divided into p displace
parallel partitions \( P_1, P_2, \ldots, P_P \). Each and every displaced partition is as big enough as it can be easily consumed in the accessible main memory because there is also a compression algorithm used.

**3. EFAR-HD ALGORITHM**

The EFAR-HD is an enhanced version of FAR-HD[4] algorithm. As we now the FAR-HD algorithm works well for high dimensional data but as the number of attributes and transactions in a database increases so there will be a more chances of misses in analysis and rules mining. Our EFAR-HD algorithm is designed to perform the research on the accuracy of the FAR-HD algorithm with smaller data sets such as patient or contact lens dataset to find out any chances of misses occurs during association rule mining, if the misses occur our algorithm will configure out and improve its performance. The algorithm is updated with fuzzy logic and we have also implemented Jaro-winkler distance algorithm to improve the performance of the algorithm.

In the theory of computer science and knowledge, the term Jaro-winkler distance [72] is a criterion for computing the sum of differentiation between two strings. The term edit distance is frequently used to refer particularly to Jaro Winkler distance. In the theory of computer science and statistics, the Jaro–Winkler distance is a extent of similarity between two strings. It is a modification of the Jaro distance metric which is a type of string edit distance and was implemented in the area of record linkage or duplicate detection. The similarity of Jaro-Winkler distance depends upon the factor that if Jaro–Winkler distance for two strings is lower than the strings are more similar. The Jaro-Winkler similarity is given by “1 - Jaro Winkler distance”. The Jaro–Winkler distance metric is premeditated and best well-matched for short strings such as somebody names. The similarity score is normalized and evaluated as 0 equates to no similarity and 1 is an exact match. In other words, The Jaro algorithm is a measure of characters in common, being no more than half the length of the longer string in distance, with consideration for transpositions. Winkler modified this algorithm to support the idea that differences near the start of the string are more significant than differences near the end of the string. Jaro and Jaro-Winkler are suited for comparing smaller strings like words and names.

Mathematically the Jaro-Winkler between two strings \( s_1, s_2 \) is given by \( d_j \) such that

\[
d_j = \left\{ \begin{array}{ll}
0 & \text{if } m = 0,
\frac{1}{m} \left( \frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m} \right) & \text{otherwise}
\end{array} \right.
\]

**Eq.(2)**

Where

- \( m \) is the number of matching characters.
- \( t \) is half the number of transpositions

Two characters from \( s_1 \) and \( s_2 \) are deliberated matching only if they are identical and not past than

\[
\left\lfloor \frac{\max(|s_1|,|s_2|)}{2} \right\rfloor - 1 \quad \text{---Eq.(3)}
\]

The number of matching but with dissimilar arrangement order characters are divided by 2 outlines the number of transpositions. For example, while comparing word “CRATE with word “TRACE”, only ‘R’ ‘A’ ‘E’ are the resulting matched characters, i.e. \( m=3 \). Although ‘C’, ‘T’ appear in both strings, they are farther than 1. i.e floor\( \frac{53}{2} - 1 = 1 \). Therefore character \( t = 0 \). In the word “DwAyNE” versus DuANE the matching characters are by this time in the similar order D-A-N-E, so there is no transpositions are needed in this case. Jaro Winkler distance uses a prefix scale \( pp \) which gives auspicious ratings to character sequences that equals from the beginning for a given prefix length set.
Given two strings are $s_1$ and $s_2$ are their winkler distance is denoted by $d_{ww}$ is given as

$$d_{ww} = d_j + (p \cdot (1 - d_j)) \text{ ......................................... Eq.(4)}$$

Where

- $d_j$ is Jaro distance for strings $s_1$ and $s_2$.
- $l$ is the length of common prefix at the start of string up to maximum of 4 characters.
- $p$ is the constant scaling factor for how much the score is adjusted upwards for having common prefixes. The value of $p$ should not exceed 0.25, otherwise the distance becomes larger than 1. The standard value for this constant in Jaro-Winkler work is $p = 1$.

In some implementations of Jaro-Winkler, the prefix bonus $lp \cdot 1-\frac{d_j}{l}$ is only needed when the compared strings have a Jaro distance above a set "boost threshold" $bt$. The boost threshold in Winkler's implementation was 0.7. Hence the winkler distance is denoted by $d_{ww}$ with boost threshold is given as $dw$

Below is the Jaro-Winkler distance algorithm pseudo code:

**Step 1: Matches:** The match phase is a greedy alignment step of characters in one string against the characters in another string. The maximum distance (measured by array index) at which characters may be matched is defined by:

$$\text{matchRange} = \max(\text{cs1.length()}, \text{cs2.length()}) / 2 - 1$$

The match phase is a greedy alignment that proceeds character by character through the first string, though the distance metric is symmetric (that, is reversing the order of arguments does not affect the result). For each character encountered in the first string, it is matched to the first unaligned character in the second string that is an exact character match. If there is no such character within the match range window, the character is left unaligned.

**Step 2: Transpositions:** After matching, the subsequence of characters actually matched in both strings is extracted. These subsequences will be the same length. The number of characters in one string that do not line up (by index in the matched subsequence) with identical characters in the other string is the number of "half transpositions". The total number of transpositions is the number of half transpositions divided by two, rounding down.

The Jaro distance is then defined in terms of the number of matching characters matches and the number of transpositions, transposes:

$$\text{jaroProximity(cs1,cs2)} = \left( \frac{\text{matches}(cs1,cs2)}{\text{cs1.length()}} + \frac{\text{matches}(cs1,cs2)}{\text{cs2.length()}} + \frac{\text{matches}(cs1,cs2) - \text{transposes}(cs1,cs2)}{\text{matches}(cs1,cs2)} \right) / 3$$

$$\text{jaroDistance(cs1,cs2)} = 1 - \text{jaroProximity(cs1,cs2)}$$

In words, the measure is the average of three values; (a) the percentage of the first string matched, (b) the percentage of the second string matched, and (c) the percentage of matches that were not transposed.

Jaro-Winkler Distance has a large variety of applications such as spell checkers, correction systems for OCR and a software tool used to help out natural language transformation based on conversion memory. The jaro-winkler distance can also be used as an aid in fuzzy string matching and searching in applications such as record linkage, the compared strings are typically short to facilitate improve speed of assessment.

Just like FAR-HD algorithm, EFAR-HD employs two phased approach, in the first phase the algorithm scans each transaction in the current partition of the dataset and find out the common candidate items, the function name build association do this task to find the out the common items in the dataset, we have the sued the
contact lens dataset and produce a tidlist for each singleton found. The Jaro-Winkler distance here checked the difference of two strings. Jaro-Winkler distance divides the dataset into two partitions. One partition contains similar strings which are short in length and other partitions contain the string who having long length. The long length strings are discarded and can’t be used for rules pruning. According to Jaro-Winkler distance the string whose Jaro-Winkler distance is more than three is long length string which can’t be used for rules pruning. After all singletons in the existing partition have been inspected, the Tidlists of singletons which are not common or whose Jaro-Winkler distance is more than three are dropped.

Rules pruning are done in the second phase during this phase association rule mining using fuzzy has been done. The algorithm one by one traverses each and every partition from the beginning and finds out the common and frequent candidate items over the whole dataset. The rules pruning are carried out on these common candidate items.

**Pseudocode of EFARM algorithm [4]**

**Phase - I:**

- navigate every partition $p_t \in P$
do
- navigate every operation $0 \in$ existing partition $p_t$ do
- for every singleton $s \in$ existing operation $0$ do
- compute $\mu$ for $s$
- If $ddjf$ for $si$ to $s < 1$
  - count[$s$] += $\mu$
- Else
- Swap syllables
- If $ddjfffffssli$ to $s < 1$
  - counts[$s$] += $\mu$
- end If
- navigate every singleton $si$ where $i = 1$ to $\mid Sp\mid$ do
- If $si$ is not d-frequent i.e (common candidate item) then
eradicate Tid[$si$]
- end if
- end for
- navigate every singleton $si$ where $i = 1$ to $\mid Sp\mid$ do
- navigate every singleton $sj$ where $j = 1$ to $\mid Sp\mid$ do
- CreateNewItemSet($si$, $sj$)
- end for
- end for

**Pseudocode to create newitemset of commom candidate items**

- CreateNewItemSet:
- Coalesce $IT$ and $sf$ to get new item $ITnew$
- Tid[$ITnew$] = Tid[$IT$] $\cap$ Tid[$ITsf$]
- Compute $\mu_p$ for $ITnew$ using tid[$ITnew$]
- count[$ITnew$] += $\mu_p$
- If $ITnew$ is common candidate item or d-frequent then
  - Navigate every singleton $sk$ where $k = f + 1$ to $\mid Sp\mid$ do
  - createNewItemSet($ITnew$, $sk$)
  - end for
- Eliminate Tid[$ITnew$]

**Phase 2**

- navigate every partition $p_t \in PP$ do
- navigate every itemset $IT \in$ pt in the first phase do
- if $IT$ is recurrent in excess of the complete dataset $EE$ then
- output $IT$
- end if
- eliminate $IT$
- end for
- for every enduring itemset $IT$ do
- classify ingredient singletons $s_1$, $s_2$, $s_3$, $\ldots$, $s_{St}$ of $IT$
such that $it = s_1 \cap s_2 \cap s_3 \cap \ldots \ldots \ldots \ldots \ldots \ldots s_{St}$
- Tidlist is $tti[IT]$ = interconnect Tidlists of each and every one essential singletons
- compute $\mu_{IT}$ for $IT$ using Ti[IT]
- $cco[IT] += \mu$
EFAR-HD is designed to work in efficient manner. Just like the previous algorithm EFAR-HD uses the same function and logic during the phase 2 and it is unchanged during the implementation as this is the updated only during the phase 1 of the previous algorithm developed by Ashish Mangalampalli and Vikram Pudi. Further in during phase 2 the algorithm computed for each residual itemset IT, discover its essential singletons $s_1, s_2, s_t$ and then attain the Tidlist of IT(Tidd(IT)) by interconnecting the Tidlists of all the ingredient singletons. Furthermore, the count up of every singleton IT is restructured in [IT]. Hence, got exchange among outputting and deleting itemsets and generating Tidlists for itemsets in anticipation of no supplementary itemsets are left behind.

3. EXPERIMENTAL SETUP AND ANALYSIS

In this section, we explain the experimental setup and analysis used for comparing EFAR-HD with two other Fuzzy ARM algorithms Fuzzy Apriori and FAR-HD - the first is the algorithm described in [9] and [10] and the second one being a . We have re-implement the FARHD and Fuzzy Apriori in the java and further the algorithm FAR-HD is enhanced by adding the Jaro-Winkler distance in the coding to check the degree of similarity or to compute the distance between two strings. Further we have introduced the concept of phonetics in the EFAR-HD which replaces the character “ee” with the “i” if required. The execution of the algorithm is carried on the eclipse kepler and connect it with the weka tool to find the associations between different items. The weka tool provides the interface to connect the dataset with the EFAR-HD algorithm.

EXPERIMENTAL RESULTS

Contact Lens Dataset. The dataset is complete which includes all probable grouping of attribute-value pairs that represents which patient is suitable with hard contact lenses and which patient is suitable soft contact lenses based on the symptoms of eye patient. This contact lens dataset which is an collection of other smaller datasets and is one the biggest dataset on which we have performed our experiment and is of the dimension of a distinctive dataset for which EFARM is intended to work best.

Fig 1. Minsupp Support from 0.0001 to 0.1

Fig 2. Minsupp Support from 0.1 to 0.4

The routine metrics in the experimentation are overall execution time and utmost memory used. As in many of the ARM investigational evaluation, overall implementation time is the key performance metric. The highest memory used includes only the memory engaged by the Tidlists and count up of itemsets and also contains the itemsets themselves which supplies the performance metric only for the assessment of EFAR-HD, FAR-HD and FARM. The experiments were performed on a computer with WINDOWS 7, Intel corei5 processor and 4 GB DDR2 RAM. Figures 3, 4 and 5 demonstrate the outcome acquired by running EFAR-HD, FARM, and FAR-HD is that the
EFAR-HD generate rules faster than FARM, and FAR-HD for minimum support values ranging from 0.0001–0.4. Figure 1 and 2 shows the graphical outcome of EFAR-HD, FAR-HD, and FARM algorithms which shows that as we increases the MinSupport EFAR-HD generates rules in less time as compared to FARM, and FAR-HD algorithm. The one key point in the output of EFAR-HD algorithm is that the same rules with different option like ‘YES’ or ‘NO’ is merged in the single rule which is not produced in the output of , FAR-HD and FARM algorithm. From the results it is clear that EFAR-HD slightly improves the FAR-HD algorithm in terms of rules pruning on the smaller datasets used and on FAR-Miner, the EFAR-HD gives more accuracy on the large high-dimensional dataset (Consolidated dataset).

As we already know that the FAR-HD algorithm developed by Ashish Mangalampalli and Vikram Pudi possesses the byte-vector demonstration of Tidlists also contains the depth first like itemset creation strategy saved in RAM in compacted form using zlib compression algorithm yields high performance. This feature is also implemented in EFAR-HD to get FAR-HD like performance.

4. CONCLUSIONS
We have presented a fresh FARM algorithm, called EFAR-HD, for the smaller and crisp datasets such that patient and sales or marketing datasets as a viable and proficient option to Fuzzy Apriori and FAR-Miner [9] and [10] designed for the smaller datasets also as this algorithm is enhanced in terms of the accuracy and fast execution. From an experiential point of view, we have tried to improve the accuracy of FAR-HD in terms of rules generation in less time on the basis of a performance metric and parameters such that minsupport. As future work, we intend to use EFAR-HD with Soundex distance or N-gram distance algorithm and check its accuracy on the similar parameters built.

6. REFERENCES
large databases,” in SIGMOD Conference, 1993, pp. 207–216.


[31] Han, J, Pei, J, Yin, Y: Mining Frequent Patterns without Candidate Generation. In: SIGMOD Conference, ACM Press, Page(s): 1-12, 2000.


www.iosrjournals.org 8


[49] J. Han and Y. Fu. "Discovery of multiple-level association rules from large databases," in the
international conference on very large databases, Zurich, Switzerland, Page(s): 420-431, 1995.


[67] Hastie, Trevor; Tibshirani, Robert; Friedman, Jerome (2009). "The Elements of Statistical Learning:

Data Mining, Inference, and Prediction”. Retrieved 2012-08-07.


[69] Ming-Syan Chen, Jiawei Han, Philip S yu. Data Mining: An Overview from a Database Perspective[J]. IEEE Transactions on Knowledge and Data Engineering, 1996.


