Volume 8, No. 4, May 2017 (Special Issue)
ISSN No. 0976-5697
International Journal of Advanced Research in Computer Science
RESEARCH PAPER
Available Online at www.ijarcs.info

Performance Analysis of Enhanced Fuzzy Association Rule Mining Algorithm with Levenstein Distance Using Contact Lens Dataset!

Dhiraj Kapila
Student PhD (C.S.E)
SBBS University
Jalandhar, Punjab, India
Email: dhirajkapila@gmail.com

***Abstract*— Association Rule Mining (ARM) with fuzzy logic perception smooth the progress of the easy process of mining of underlying frequent or recurrent patterns based on their own frequencies in the form of association rules from any transactional and relational datasets containing items to signify 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 automate decision-making. In the concept of data mining, generally fuzzy Association Rule Mining (FARM) technique has been comprehensively adopted in transactional and relational datasets those datasets containing items who has 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. Hence, in this paper we propose E-FAR-HD algorithm which is an enhanced version of FAR-HD algorithm that designed exclusively for large or high-dimensional datasets. We have designed 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 Cluster- ing, Fuzzy Partitioning, Fuzzy Relations, Partitions, Tidlists, High Dimensions, Large Datasets, Smaller Datasets,**

I. INTRODUCTION

Data mining is the technique to dig out the inherent information and knowledge from the collection of, incomplete, imperfect ,noisy, fuzzy, random and unsystematic data which is potentially functional and people do not know in advance about this hidden information [65]. The important difference between the traditional data analysis technique such as query reporting and the data mining and is that the data mining is very helpful to determine knowledge and also useful in mining information based on the premise of no clear hypothesis [66]. The most important use of data mining is in programmed data analysis technique to come across or to find out earlier unseen or undiscovered associations among various data items in the dataset.

Data mining is the complete 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 consisting of discovering meaningful and hidden patterns and information in large datasets of items. The applications of data mining involving methods of intersection of artificial intelligence, machine learning, statistics, and database systems. [70] The overall aim of the data mining procedure is to dig out meaningful and hidden information from a dataset containing items and then renovate it into a reasonable structure for future use. [70] Apart from data analysis step, it also involves the concepts of database and data management, data pre-processing. Various other activities 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 imperative research area in the concept of data mining that facilitate the mining of concealed recurrent patterns that based on their own frequencies in the shape of association rules from any itemset or datasets containing entities to represent 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 (associative classification [25], [26], [27]) and collecting (ARM- driven clustering, like document clustering [28], [29], [20], [31]) which helps domain area experts to automate decision-making solutions. Now a day's FARM has achieve tremendous 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

CONFERENCE PAPERS
National Conference on Emerging Trends on Engineering & Technology (ETET-2017)
On 21st April 2017
University Inst. of Engg. & Tech. & University Inst. of Computer, SBBS University, Punjab (India)

have also work for low dimensional datasets are yet to be proven out.

II. FAR-HD ALGORITHM

FAR-HD algorithm has been proposed and developed by Ashish Mangalampalli and Vikram Pudi [36] 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 [36], Ashish Mangalampalli and Vikram Pudi has mentioned about the fuzzy pre-processing strategy and fuzzy actions that are employed for the authentic FARM process. 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

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} \mu_{ij}{}^{m} \left\| x_i - y_j \right\|^2 \quad \text{……………..Eq.(1)}$$

Where m is any real number in the range such that $1 \leq m < \infty$, and $\mu_{i,j}$ is the degree of membership of x_i in the cluster of j, x_i is the i^{th} dimensional measured data, c_i is the d-dimensional cluster center, and ‖ * ‖ is any norm stating the comparison between any measured data and the center. The variable m is known as fuzziness parameter which is an arbitrary real number ($m > 1$). 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 m (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 (μ). 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 0.1for the membership function μ 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 p disjoint flat separations P_1, P_2,..... 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
- P = Set of separations
- S_p = Set of singletons in existing separation p
- $td[it]$ = Tidlist of itemset it
- μ_p = collective fuzzy membership or fuzzy support of any itemset in existing partition p
- $count[it]$ = collective μ of itemset it over all separations p in which it has been executed
- d = number of partitions for some exacting itemset $i$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 i^{th} cell of the byte-vector includes the μ for the i^{th} 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 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 itemset, and generates a Tidlist 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 d-frequent or not, the Tidlists having singletons who are seems not to be d-frequent are dropped out. The formation of Tidlist is carried out very soon as the new data set has been created. An itemset is said to be d-frequent if its incidence over d partitions are equals to or surpasses the support accustomed for d partitions, then the itemset is considered to be frequent over d partitions of the dataset E . Further the authors explained that the calculation of each and every singleton s is preserved in the array data structure $[s]$. To produce the bigger itemsets, they the use depth-first search (DFS) traversing approach, i.e. starts with a singleton s_i and create all the supersets of s_i, prior to doing the similar for the subsequently singleton s_i+1. Primarily, every singleton s_i is united with one more singleton s_j to create supersets of s_i in depth-first search manner. This progression is completed for each, s_j where $j = i + 1$ to | S_p |. 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 E are the output. This output dataset E is further logically divided into p displace parallel partitions P_1,P_2,….,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.

III. EFAR-HD ALGORITHM

The EFAR-HD is enhanced version of FAR-HD 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 levenstein distance algorithm to improve the performance of the algorithm

In the theory of computer science and knowledge, the term Levenshtein 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 Levenshtein distance. The Levenshtein distance between two strings is defined as the lowest number of edits necessary to convert one string into the other with the acceptable edit operations such as insertion, deletion, or substitution of a single character is allowed. It is named after Vladimir Levenshtein, who considered this distance in 1965. In other words, Levenshtein distance (LD) is a measure of the resemblance between two strings, the source string (s) and the target string (t). Therefore according to Vladimir Levenshtein, the distance is the minimum number of deletions, insertions, or substitutions required to transform source string (s) and the target string (t). The higher is the Levenshtein distance, the more different the strings are.

Mathematically the Levenshtein distance between two strings a, b is given by $\mathrm{lev}_{a,b}(|a|,|b|)$ such that

$$\mathrm{lev}_{a,b}(|a|,|b|) = \begin{cases} \max(i,j) & \text{if } \min(i,j)=0, \\ \min\begin{cases} \mathrm{lev}_{a,b}(i-1,j)+1 \\ \mathrm{lev}_{a,b}(i,j-1)+1 \\ \mathrm{lev}_{a,b}(i-1,j-1)+1_{(a_i\neq b_j)} \end{cases} & \text{otherwise} \end{cases} \quad \ldots \quad \text{....Eq.(2)}$$

For example, the Levenshtein distance between the two strings "kitten" and "sitting" is 3, here “kitten is the source string (s) and "sitting" is target string (t) “since the following three edits are required to transform string (s) and the target string (t), and there is no way to do it with fewer than three edits:

a. kitten → sitten (substitution of 's' for 'k')
b. sitten → sittin (substitution of 'i' for 'e')
c. sittin → sitting (insertion of 'g' at the end).

Below is the Levenstein distance algorithm pseudo code:

Step 1: Initialization
a. Set n to be the length of s, set m to be the length of t.
b. Construct a matrix containing 0……..m rows and 0………n columns.
c. Initialize the first row to 0………..n.
d. Initialize the first column to 0…………….m.

Step2: Processing
a. Examine s (i from 1 to n).
b. Examine t (j from 1 to m).
c. If s[i] equals t[j], then cost is 0.
d. If s[i] doesn't equal t[j], then cost is 1.
e. Set cell d[i,j] of the matrix equal to the minimum of:
i. The cell immediately above plus 1: d[i-1,j] + 1.
ii. The cell immediately to the left plus 1: d[i,j-1] + 1.
iii. The cell diagonally above and to the left plus the cost: d[i-1,j-1] + cost.

Step 3: Result

Step 2 is repeated till the d[n,m] value is found

Levenstein Distance has [74] 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 Levenshtein 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 levenstein distance here checked the difference of two strings. Levenstein 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 levenstein distance the string whose levenstein 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 levenstein 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 [36]

Phase - I:

- navigate every partition $p_t \in P$ do
- navigate every operation 0 ∈ existing partition p_t do
- for every singleton s ∈ existing operation 0 d o
- compute µ for every s
- If Ld for s_i to s <1
- count[s]+=µ
- Else
- Swap syllables
- If Ld sli to s<1
- counts[s]+=µ
- end If end for end for
- navigate every singleton s_i where i=1 to $|S_p|$ do
- If s_i is not d-frequent i.e (common candidate item) then eradicate Tid[s_i]
- end if
- end for
- navigate every singleton s_i where i=1 to $|s_p|$ do
- navigate every singleton s_j where j=1 to $|s_p|$ do
- CreateNewtItemSet(s_i,s_j)
- end for end for end for

Pseudocode to create newitemset of commom candidate items

- CreateNewItemSet:
- Coalesce IT and s_f to get new item IT_{new}
- Tid[IT_{new}]=Tid[IT]∩Tid[ITsf]
- Compute μ_p for IT_{new} using tid[IT_{new}]
- count[IT_{new}]+=μ_p
- If IT_{new} is common candidate item or d-frequent then
- Navigate every singleton s_k where k=f+1 to $|s_p|$ do
- createNewItemSet(IT_{new},s_k)
- end for end if
- Eliminate Tid[IT_{new}]

Phase 2

- navigate every partition $p_t \in P$ do
- navigate every itemset IT ∈ p_t in the first phase do
- if IT is recurrent in excess of the complete dataset *E* then
- output IT
- end if
- eliminate IT
- end for
- for every enduring itemset IT do
- classify ingredient singletons $s_1, s_2, \ldots, s_t$ of IT such that $it = s_1 \cap s_2 \cap s_3 \cap \ldots \ldots \ldots \ldots \ldots s_t$
- Tidlist is tid[IT] = interconnect Tidlists of each and every one essential singletons
- compute μ for IT using Tid[IT]
- $count$[IT]+ = μ
- end for
- if no itemsets stay behind to be count then exit
- end if
- end for

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(Tid[IT]) by interconnecting the Tidlists of all the ingredient singletons. Furthermore, the count up of every singleton IT is restructured in c[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 levenstein 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.

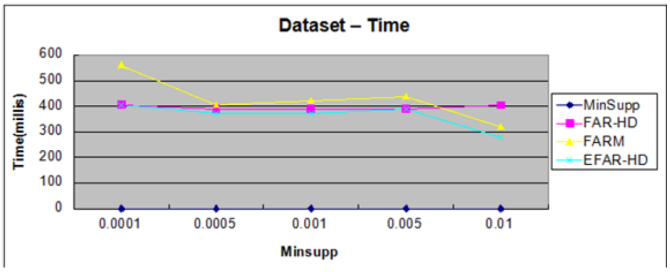

Fig 1. Minsupp Support from 0.0001 to 0.1

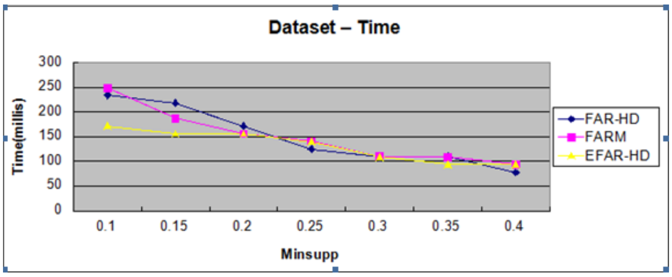

Fig 2. Minsupp Support from 0.1 to 0.4

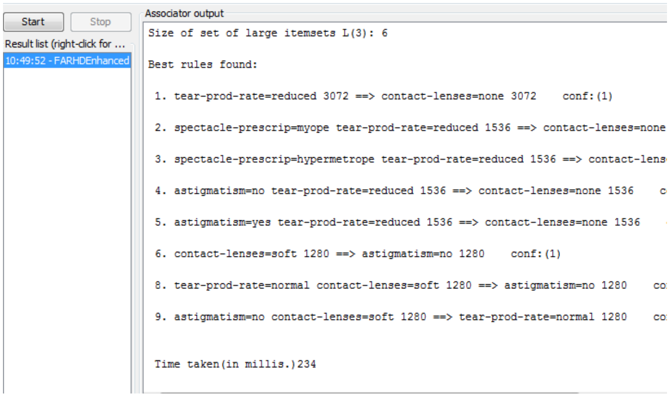

Fig 3. Rules generated by EFARHD in Milisec

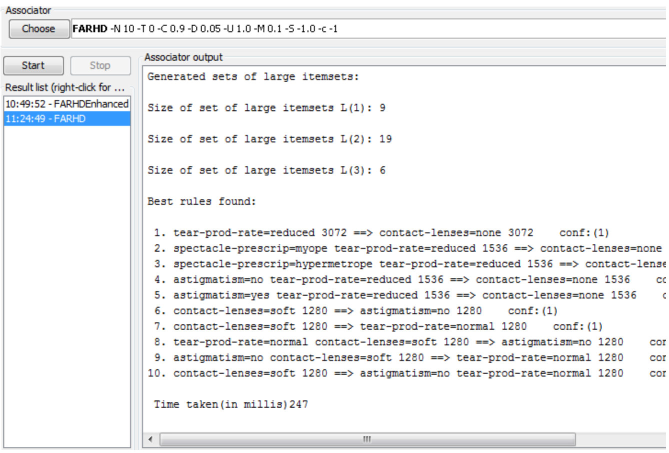

Fig 4. Rules generated by FAR-HD in milisec

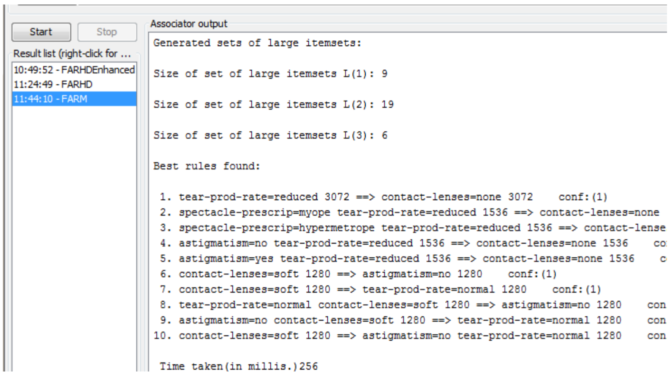

Fig 5. Rules generated by FARM in milisec

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 shoes 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.

IV. 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 Jaro Winkler algorithm and check its accuracy on the similar parameters built.

REFERENCES

[1] X. Yin and J. Han, "CPAR: Classification based on predictive association rules," in SDM, 2003.

[2] F. A. Thabtah, "A review of associative classification mining," Knowledge Eng. Review, vol. 22, no. 1, pp. 37–65,

[3] Veloso, W. M. Jr., and M. J. Zaki, "Lazy associative classification," in ICDM, 2006, pp. 645–654.

[4] L.Zhuang and H. Dai, "A maximal frequent itemset approach for web document clustering," in CIT, 2004, pp. 970–977.

[5] H. Yu, D. Searsmith, X. Li, and J. Han, "Scalable construction of topic directory with nonparametric closed termset mining," in ICDM, 2004, pp. 563–566.

[6] B. C. M. Fung, K. Wang, and M. Ester, "Hierarchical document clustering using frequent itemsets," in SDM, 2003.

[7] H. H. Malik and J. R. Kender, "High quality, efficient hierarchical document clustering using closed interesting itemsets," in ICDM, 2006, pp. 991–996.

[8] Pudi and J. R. Haritsa, "ARMOR: Association rule mining based on ORacle," in FIMI, 2003.

[9] Mangalampalli and V. Pudi, "FAR-miner: a fast and efficient algorithm for fuzzy association rule mining," IJBIDM, vol. 7, no. 4, pp. 288–317, 2012.

[10] Mangalampalli and V. Pudi, "Fuzzy association rule mining algorithm for fast and efficient performance on very large datasets," in FUZZ-IEEE, 2009, pp. 1163–1168.

[11] R. Agrawal, T. Imielinski, and A. N. Swami, "Mining association rules between sets of items in large databases," in SIGMOD Conference, 1993, pp. 207–216.

[12] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules in large databases," in VLDB, 1994, pp. 487–499.

[13] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," in SIGMOD Conference, 2000, pp. 1–12.

[14] P. Yan, G. Chen, C. Cornelis, M. D. Cock, and E. E. Kerre, "Mining positive and negative fuzzy association rules," in KES, 2004, pp. 270–276.

[15] M. D. Cock, C. Cornelis, and E. E. Kerre, "Elicitation of fuzzy association rules from positive and negative examples," Fuzzy Sets and Systems, vol. 149, no. 1, pp. 73–85, 2005.

[16] H. Verlinde, M. D. Cock, and R. Boute, "Fuzzy versus quantitative association rules: A fair data-driven comparison," IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics, vol. 36, no. 3, pp. 679–683, 2005.

[17] E. H¨ullermeier and Y. Yi, "In defense of fuzzy association analysis," IEEE Transactions on Systems, Man, and Cybernetics, Part B, vol. 37, no. 4, pp. 1039–1043, 2007.

[18] M. D. Cock, C. Cornelis, and E. E. Kerre, "Fuzzy association rules: A two-sided approach," in FIP, 2003, p. 385390.

[19] D. Dubois, E. H¨ullermeier, and H. Prade, "A systematic approach to the assessment of fuzzy association rules," Data Min. Knowl. Discov., vol. 13, no. 2, pp. 167–192, 2006.

[20] D. Dubois, E. H¨ullermeier, and H. Prade, "A note on quality measures for fuzzy association rules," in IFSA, 2003, pp. 346–353.

[21] M. Delgado, N. Marn, D. S´anchez, and M. A. V. Miranda, "Fuzzy association rules: General model and applications," IEEE Transactions on Fuzzy Systems, vol. 11, pp. 214–225, 2003.

[22] J. C. Dunn, "A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters," Journal of Cybernetics, vol. 3, pp. 32–57, 1973.

[23] J. C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms. Norwell, MA, USA: Kluwer Academic Publishers, 1981.

[24] H. Bay, A. Ess, T. Tuytelaars, and L. J. V. Gool, "Speeded-up robust features (SURF)," Computer Vision and Image Understanding, vol. 110, no. 3, pp. 346–359, 2008

[25] Fayyad, Usama; Piatetsky-Shapiro, Gregory; Smyth, Padhraic: The KDD Process for Extracting Useful Knowledge from Volumes of Data. Communications of the ACM, Volume 39, Issue 11, Page(s): 27 – 34, 1996.

[26] http://www.umsl.edu/~joshik/msis480/chapt11.htm

[27] J. Han and M. Kamber, Data Mining: Concepts and Techniques: The Morgan Kaufmann Series, 2001.

[28] [Hipp, Jochen; Guentzer, Ulrich; Nakhaeizadeh, Gholamreza: Algorithms for Association Rule Mining - A General Survey and Comparison. ACM SIGKDD Explorations Newsletter, Volume 2, Issue 1, 2000.

[29] Agrawal, Rakesh; Imielinski, Tomasz; Swami, Arun: Mining Association Rules between Sets of Items in Large Databases. Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, 1993.

[30] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules in Large Databases", Proc. of the 20th International Conference on Very Large Data Bases, VLDB, Page(s): 487-499, 1994.

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

[32] Türks¸en, I.B. and Tian Y. 1993. Combination of rules and their consequences in fuzzy expert systems, Fuzzy Sets and Systems, No. 58,3-40, 1993.

[33] http://www.cs.cmu.edu/Groups/AI/html/faqs/ai/fuzzy/part1/faq-doc-4.html

[34] L.A.Zadeh.Outline of a new approach to the analysis of complex systems and decision processes. IEEE Transactions on System, Man, and Cybernetics, Volume 3, Pages(s):28-44, January, 1973.

[35] Wai-HO AU, Keith C.C. Chan: An Effective Algorithm for Discovering Fuzzy Rules in Relational Databases, Fuzzy Systems Proceedings, IEEE World Congress on Computational Intelligence. Volume 2.ISSN: 1098-7584, Print ISBN: 0-7803-4863-X, Page(s):1314 – 1319, 1998.

[36] Ashish Mangalampalli, Vikram Pudi: FAR-HD: A Fast And Efficient Algorithm For Mining Fuzzy Association Rules In Large High-Dimensional Datasetss. FUZZ-IEEE 2013,.

[37] Zadeh, L. A.: Fuzzy sets. Inf. Control, 8, Page(s): 338–358, 1965.

[38] Borgelt, Christian: An Implementation of the FP-growth Algorithm. ACM Press, New York, NY, USA, 2005. A Survey on Fuzzy Association Rule Mining Methodologies

[39] www.iosrjournals.org 8

[40] Pudi, V., Haritsa, J.: ARMOR: Association Rule Mining based on Oracle. CEUR Workshop Proceedings, 90, 2003.

[41] Savasere, A., Omiecinski, E., Navathe, S.B.: An Efficient Algorithm for Mining Association Rules in Large Databases. In: VLDB, Morgan Kaufmann, Page(s): 432-444, 1995.

[42] Dunn, J. C.: A Fuzzy Relative of the ISODATA Process and its Use in Detecting Compact Well Separated Clusters. J. Cybernetics and Systems,Volume 3,Page(s):32-57, 1974.

[43] Bezdek, J. C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Kluwer Academic Publishers, Norwell, MA, 1981.

[44] Hoppner, F., Klawonn, F., Kruse, R, Runkler, T.: Fuzzy Cluster Analysis, Methods for Classification, Data Analysis and Image Recognition. Wiley, New York, 1999.

[45] De Cock, M., Cornelis, C., Kerre, E.E.: Fuzzy Association Rules: A Two-Sided Approach. In: FIP, Page(s): 385-390, 2003.

[46] Yan, P., Chen, G., Cornelis, C., De Cock, M., Kerre, E.E.: Mining Positive and Negative Fuzzy Association Rules. In: KES, Springer, Page(s): 270-276, 2004.

[47] Verlinde, H., De Cock, M., Boute, R.: Fuzzy Versus Quantitative Association Rules: A Fair Data-Driven Comparison. IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics, Volume 36, Page(s): 679-683, 2006.

[48] Ehsan Vejdani Mahmoudi, Vahid Aghighi, Masood Niazi Torshiz, Mehrdad Jalali, Mahdi Yaghoobi: Mining generalized fuzzy association rules via determining minimum supports ,IEEE Iranian Conference on Electrical Engineering (ICEE)2011, E-ISBN :978-964-463-428-4 ,Print ISBN:978-1-4577-0730-8,Page(s):1 – 6, 2011.

[49] J. Han, et al., "Mining top-k frequent closed patterns without minimum support, " In Proceedings of the 2002 IEEE international conference on data mining, Page(s): 211- 218, 2002.

[50] 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.

[51] T. P. Hong, et al., "An ACS-based framework for fuzzy data mining," Expert Systems with Applications, Volume 36, Page(s): 11844-11852, Nov, 2009.

[52] T. P. Hong, et al., "Mining fuzzy multiple-level association rules from quantitative data," Applied Intelligence, Volume 18

[53] , Page(s): 79-90, Jan-Feb, 2003.

[54] Y. C. Lee, et al., "Multi-level fuzzy mining with multiple minimum supports," Expert Systems with Applications, Volume 34,Page(s): 459-468, Jan, 2008.

[55] Toshihiko Watanabe: Fuzzy Association Rules Mining Algorithm Based on Output Specification and Redundancy of Rules, IEEE International Conference on Systems, Man, and Cybernetics (SMC) 2011, ISSN: 1062-922X, Print ISBN: 978-1-4577-0652-3, Page(s):283 – 289, 2011.

[56] Y. C. Lee, T. P. Hong, and T. C. Wang, “Mining Fuzzy Multiple-level Association Rules under Multiple Minimum Supports,” Proc. of the 2006 IEEE International Conference on Systems, Man, and Cybernetics, Page(s): 4112-4117, 2006.

[57] T. Watanabe: “An Improvement of Fuzzy Association Rules Mining Algorithm Based on Redundancy of Rules,” Proc. of the 2nd International Symposium on Aware Computing, Page(s): 68-73, 2010.

[58] M. Delgado, N. Marin, M. J. Martin-Bautista, D. Sanchez, and M.-A.Vila, “Mining Fuzzy Association Rules: An Overview,” Studies in Fuzziness and Soft Computing, Springer, Volume 164/2005, Page(s): 351-373, 2006.

[59] M. Delgado, N. Marin, D. Sanchez, and M.-A. Vila, “Fuzzy Association Rules: General Model and Applications,” IEEE Trans. on Fuzzy Systems, Volume 11, No.2, Page(s): 214-225, 2003.

[60] Y. Xu, Y. Li, and G. Shaw, “Concise Representations for Approximate Association Rules,” Proc. of the 2008 IEEE International Conference on Systems, Man, and Cybernetics, Page(s): 94-101, 2008.

[61] UCI Machine Learning Repository: http://www.ics.uci.edu/~mlearn/MLRepository.html

[62] Toshihiko WATANABE, Ryosuke Fujioka: Fuzzy Association Rules Mining Algorithm Based on Equivalence Redundancy of Items, IEEE International Conference on Systems, Man, and Cybernetics (SMC), 2012, E-ISBN: 978-1-4673-1712-2, Print ISBN: 978-1-4673-1713-9, Page(s):1960 – 1965, 2012.

[63] Frawley, William J.; Piatetsky-Shapiro, Gregory; Matheus, Christopher J.: Knowledge Discovery in Databases: an Overview. AAAI/MIT Press, 1992.

[64] Delgado, Miguel: Fuzzy Association Rules: an Overview. BISC Conference, 2003.

[65] Pawlak. Z. Rough Sets International Journal of Computer and Information Sciences, Page(s):341-356, 1982.

[66] Agrawal, R., Imielinski, T., and Swami, A. N. 1993. Mining association rules between sets of items in large databases. In Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, 207-216.

[67] Fayyad, Usama; Piatetsky-Shapiro, Gregory; Smyth, Padhraic (1996). "From Data Mining to Knowledge Discovery in Databases". Retrieved 17 December 2008.

[68] "Data Mining Curriculum". ACM SIGKDD. 2006-04-30. Retrieved 2011-10-28.

[69] Clifton, Christopher (2010). "Encyclopædia Britannica: Definition of Data Mining". Retrieved 2010-12-09.

[70] Hastie, Trevor; Tibshirani, Robert; Friedman, Jerome (2009). "The Elements of Statistical Learning: Data Mining, Inference, and Prediction". Retrieved 2012-08-07.

[71] Shilpa N. Ingoley, J.W. Bakal,”Evaluating Students’ Performance using Fuzzy Logic” in International Conference in Recent Trends in Information Technology and Computer Science (ICRTITCS - 2012) Proceedings published in International Journal of Computer Applications

[72] Ming-Syan Chen, Jiawei Han, Philip S yu. Data Mining: An Overview from a Database Perspective[J]. IEEE Transactions on Knowledge and Data Engineering, l996,

[73] R Agrawal ,T 1 mielinski, A Swami. Database Mining: A Performance Perspective[J]· IEEE Transactions on Knowledge and Data Engineering, 1993,12:914-925.

[74] Y. Peng, G. Kou, Y. Shi, Z. Chen (2008). "A Descriptive Framework for the Field of Data Mining and Knowledge Discovery" *International Journal of Information Technology and Decision Making, Volume 7, Issue 4* 7: 639 – 682. doi:10.1142/S0219622008003204.

[75] S Schimke, C Vielhauer, J Dittmann. Using Adapted Levenshtein Distance for On-Line Signature Authentication. Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04), 2004.

[76] S Schimke, C Vielhauer. Similarity searching for on-line handwritten documents. - Journal on Multimodal User Interfaces, 2007 –

[77] Springer [74] Li Yujian, Liu Bo, A Normalized Levenshtein Distance Metric, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 6, pp. 1091-1095, June, 2008.