PERFORMANCE EVALUATION OF E-FARHD ALGORITHM WITH LEVENSTEIN DISTANCE ON SOYBEAN DATASET

Dhiraj Kapila[1], Dr Vinay Chopra[2]
[1]Student M.Tech (C.S.E), [2]Asst.Professor (C.S.E)
D.A.V Institute of Engg & Technology
Jalandhar, Punjab, India

Abstract: Association Rule Mining (ARM) with fuzzy logic concept make possible the easy progression of drawing out of latent repeated or recurrent patterns based on their individual frequencies in the outline of association rules from several transactional and relational datasets including items to imply the most topical trends in the specified dataset. These extract recurring patterns or fuzzy association rules make use of either for physical data analysis or also inclined to force further mining tasks like categorization or classification commonly known as associative classification and collecting or clustering generally ARM- driven clustering, like document clustering is employed which helps out province area professionals to mechanize decision-making clarifications. In the conception of data mining, commonly fuzzy Association Rule Mining (FARM) practices has been expansively taken up in transactional and relational datasets, those datasets containing items who has a smaller quantity to medium quantity of attributes/dimensions. A small number of procedures have also accepted for high dimensional dataset also, but whether those procedures have also work for low dimensional datasets are yet to be verified out. Hence, in this paper we recommend E-FAR-HD algorithm which is an improved edition of FAR-HD algorithm that designed and developed entirely for large or huge dimensional datasets. We have designed and developed this EFAR-HD algorithm that enlarges the accurateness of FAR-HD algorithm on the slighter datasets and eradicate the chances of misses when FAR-HD has experienced on smaller datasets such as soybean or patient dataset.

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

I. INTRODUCTION

Data mining is the practice to pull out the intrinsic information and facts from the compilation of; deficient, deficient, noisy, fuzzy, arbitrary and disorganized data which is potentially serviceable and the public users do not be acquainted with in advance about this unseen information [65]. The key difference between the conventional data analysis practice such as query detailing and the data mining and is that the concept of data mining is very supportive to find out hidden knowledge and also helpful in drawing out information based on the assertion of no clear hypothesis [66]. Nearly all significant use of data mining is in involuntary data analysis practice to come across or to discovering out former undetected or undiscovered relations amongst various data items in the datasets. Data mining is the absolute examination step of the "Knowledge Discovery in Databases" process, or KDD),[45] which is an inter-corrective sub domain of computer science,[70][65][71] which is not anything but a computational doings consisting of finding out significant and hidden patterns and information that having an important effect in large datasets of items. The applications of data mining are concerning with the techniques of juncture of artificial intelligence, machine learning, statistics, and database systems. [70] The overall aspire of the data mining course of action is to dig out consequential and out of sight information from a dataset containing items and then refurbish it into a realistic structure for upcoming use. [70] Apart from data analyzing function, it also engages the conception of database and data management, data pre-processing. Various other activities like deduction and difficulty considerations, interestingness metrics, and post-processing of exposed arrangements are also the measurement of data mining progression. Association Rule Mining (ARM) is solitary of the most crucial exploration area in the conception of data mining that smooth the progress of the drawing out of concealed persistent patterns that based on their individual frequencies in the outline of association rules from several itemset or datasets containing units to characterizes the most topical trends in the known dataset. These mined recurrent patterns or fuzzy association rules make use of either for substantial data analysis or also inclined to force auxiliary mining tasks like categorization or classification commonly known as associative classification [25], [26], [27] and collecting or clustering generally ARM- driven clustering, like document clustering is employed [28], [29], [20], [31] which helps out province area professionals to mechanize decision-making clarifications. Now a day’s FARM has accomplished remarkable acknowledgment because of its appropriateness or precision, which can be approved to its capability to dig for large amounts of data from massive operational and relational datasets containing items. Now frequent patterns keep hold of all the existing associations between items and entities in the specified dataset and concordat only with the numerically significant relations, classification or clustering. Association rules mining (ARM) practice in extensively employed in various [40] domains such as financial data analysis, bionetwork analysis, telecommunication networks, stock market research and risk management, inventory control etc. The traditional Apriori algorithm is employed for drawing out recurrent item set by means of association rules over the operational databases. This conventional apriori algorithm is ensues by make out the repeated individual items in the database and intensifying them to superior and bigger item sets as long as those item sets come into sight effectively over and over again in the database.

Association rule mining [64] is used to locate and dig out association rules that fulfill the pre-defined minimum support and confidence from a specified dataset of items. In the conception of ARM, in general fuzzy Association Rule Mining (FARM) practice has been expansively accepted in those transactional and relational datasets containing items that have a smaller number to medium number of attributes/dimensions. Few FARM practices have also accepted for high dimensional datasets, but whether those practices have also works well for low dimensional datasets are thus far to be confirmed.

II. FAR-HD ALGORITHM

FAR-HD algorithm has been anticipated and made by Ashish Mangalampalli and Vikram Pudi [36] which is capable to excavate or dig out fuzzy association rules from HDD (high dimensional datasets). As we already aware that the conventional ARM algorithms like apriori and FP-growth look forward for twofold attributes and also these conservative ARM algorithms cannot be useful directly on those item sets and in those domains or fields, in which there is enormous quantity of involvement of numerical attributes or also have data with very huge total of numerical dimensions like representation datasets have. The image sphere dataset holds the feature vectors with more than 60 dimensions which calls for the competent and quick-witted algorithm that can intelligent to extract superior association rules from datasets or to carry out the acts like associative classification from this image dataset rapidly. So FAR-HD algorithm is one of the excellent selections to bring into play to dig for fuzzy association rules from these high-dimensional datasets. This is an resourceful algorithm which can capable to excavate fuzzy association rules from very high-dimensional geometric datasets that restrains more than 0.5 million vectors and every vector length consists of minimum 60 dimensions and analogous to the draw round of fuzzy features. FAR-HD algorithm [36] utilizes fuzzy C-means (FCM) clustering scheme to spawn fuzzy clusters from the specified feature vectors of the specified itemsets. Each feature vector will be robust in to all of the k clusters with a clear-cut stage of membership which facilitates in dipping the trouble of polysemy and synonymy which generally generates in the case of crisp clustering. The high-flying features of FAR-HD are that the algorithm personifies a two segment dealing out technique, and a Tidlist proposal for maneuvering the frequency of item sets and also utilizes a distracted Zlib compression algorithm to compact Tidlist while handing out them in order to accumulate lots of more Tidlist in the identical quantity of memory that is owed or available to store up them. Besides for, item set production and dispensation, this FAR-HD works sound in DFS like approach in order to contract with those high dimensional datasets who have produced association rules with a lot of items and their standard rule time-span is very far above the ground. Further in this paper [36], Ashish Mangalampalli and Vikram Pudi has point out the significant conception about the fuzzy pre-processing approach and fuzzy measures that are employing for the dependable FARM process. This preprocessing policy accomplished in two steps. In this first step, there is a construction of fuzzy clusters from the statistical vectors and during the subsequent step there is a conversion of crisp dataset that encloses statistical vectors into fuzzy datasets with the aid of fuzzy-cluster-based demonstration. The intention of this FAR-HD algorithm in fuzzy pre-processing approach is to curtail the following equation

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

Where m is any real number in the assortment range such that $1 \leq m < \infty$, and $\mu_{i,j}$ is the degree of relationship of x_i in the cluster of j, x_i is the i^{th} dimensional calculated data, c_i is the d-dimensional cluster center, and $\| * \|$ is any median stating the similarity between any calculated data and the center. The variable m is acknowledged as fuzziness constraint which is an capricious real number such that ($m >$ 1). The extent of fuzziness and Gaussian situation of fuzzy sets can be constrained via a proper estimated assessment under the range of 1.1–1.5 of the fuzziness constraint m (Eq. 1). Because of this rationale, the resulting fuzzy separations of the dataset are created where each evaluation of numeric elements are absolutely predictable by their relationship functions (μ). Based upon the quantity of fuzzy separations have portrayed for an element, each and every reachable crisp data is transformed to compound fuzzy data. This translation may show the way to the flexibility of combinatorial flare-up of creation of fuzzy records. So the authors have positioned a low threshold assessment which is equivalent to 0.1for the relationship function μ to manage this frequent fabrication of fuzzy records. All the way through the FARM process, the original crisp dataset is engorged with aspect values within the range of (0, 1) because the colossal amount of fuzzy severances are being done on every quantitative element. To carry out this engorged fuzzy dataset, a few trials are requisite which are based on the expression t-norms [43], [44], [45]. Due to this t-norm, the new innovative fuzzy dataset E is produced, upon this new fuzzy dataset E the intended algorithm will work. As already discussed above, the FAR-HD algorithm has makes use of two segments in a splitting up strategy to fabricate fuzzy association rules. The fuzzy dataset E is prudently alienated into p dislodge flat separations $P1$, $P2$… Each severance is as gigantic as it can easily accommodate in the reachable main memory. The authors have proposed the subsequent notations in this FAR-HD algorithm,

- E = Fuzzy dataset based upon fuzzy-cluster-based demonstration formed later than fuzzy pre-processing
- P = Set of severances
- Sp = Set of singletons in accessible severance p

- $td[it]$ = Tidlist of itemset it
- μp = Communal fuzzy relationship or fuzzy support of any itemset in obtainable severance p
- $count[it]$ = Communal μ of itemset it above all severances p in which it has been completed
- d = number of severances for some demanding itemset it has been completed since the severance in which it was inserted.

FAR-HD algorithm configuration make use of a byte-vector like data illustration in which every cell collects μ of the itemset correspondent to the cell pointer of the tid to which the μ relates. Thus, the ith cell of the byte-vector consists of the μ for the ith tid. If a rigorous operation method does not enfold the itemset under apprehension, then the cell correspondent to that operation process has assigned a 0 value. The entire byte vectors in the cell have packed together by using the well prepared compression algorithm name Zlib, preceding before to be accumulates in the memory. In this technique, they have put on a gigantic main memory space accessible at its clearance to speed up the finishing and performance of this FAR-HD algorithm. As discussed earlier this FAR-HD algorithm utilizes two-segmented practice, during the very premature in the first phase the steps of FAR-HD Algorithm scrutinize each and every operation in the obtainable severance of the itemset, and creates a Tidlist for every singleton initiates. When all singletons in the accessible separation have been programmed or generated then the test out is prepared to constitute out which singleton is d-frequent or not, the Tidlists containing singletons that are appearing not to be d-frequent are plummet out. The creation of Tidlist is taking out very shortly as the fresh dataset has been formed. An itemset is said to be d-frequent if it's prevalence over d separations are equivalent to or surpasses the support adapted for d separations, then the itemset is believed to be recurrent over d severances of the dataset E. Further the authors mentioned that the computation of each and every singleton s is conserved in the array data structure $[s]$. To fabricate the superior itemsets, they make use of depth-first search (DFS) navigating technique, i.e. initiate with a singleton si and form all the supersets of si, preceding before to doing the comparable for the consequently singleton si+1. Above all, every singleton si is combined with one or added singleton sj to produce supersets of si in depth-first search traversing mode. This progression is accomplished for every, sj where j = i + 1 to ❘ Sp ❘.Throughout the subsequent phase, all the itemsets that has been attached in the obtainable separation in the original phase are moreover have been précised over the whole dataset E, and hence may be detached or dropped out. From these detached data itemsets, those itemsets restraining singletons which are d-frequent above the intact dataset E are the output itemsets. This output dataset E is in advance sensibly divided into p dislocate equivalent separations P1,P2,….,PP. Each and all dislocates partition is as gigantic enough as it can be effortlessly devoted in the reachable main memory because there is also a zlib compression algorithm employed before.

III. EFAR-HD ALGORITHM

The EFAR-HD algorithm is the improved edition of FAR-HD algorithm. As we already know that the FAR-HD algorithm works fine for HDD (high dimensional dataset) but as the amount of elements and connections in a database enlarges so there will be a supplementary probability of misses in analysis and rules mining. Our EFAR-HD algorithm is developed and implemented as an intention to perform the exploration on the accurateness of the FAR-HD algorithm with smaller data sets such as patient datasets, soybean datasets or contact lens dataset to come across out any chance of misses occurs during association rule mining, if the misses take places during association rule mining, our algorithm will configure out and improve its performance using fuzzy logic. The algorithm is restructured with the use fuzzy logic and we have also put into operation the concept of levenstein distance algorithm to get better the routine and performance of the FAR-HD algorithm. Now in the theory of computer science and knowledge, the word Levenshtein distance [72] is a principle for calculating the sum of segregation or differentiation among two string sequences. The word edit distance is recurrently employed to refer mostly to Levenshtein distance. The Levenshtein distance among two string sequences is described as the smallest number of amendments essential to alter one string sequence into the other string sequence with the satisfactory amends operations such as insertion, deletion, or substitution of a solitary character is permitted. The Levenshtein distance is named after the computer scientist Vladimir Levenshtein, who developed this distance logic in the year1965. In other words, Levenshtein distance (LD) is a gauge of the similarity between two string sequences, the resource string sequence (s) and the target object string sequence (t). For that reason according to Vladimir Levenshtein, the edit distance is the lowest amount of deletions, insertions, or substitutions mandatory to convert resource string (s) and the target object string sequence (t). The superior is the Levenshtein distance, the extra different the strings sequences are. Precisely the Levenshtein distance amongst two strings sequences a, b is given by leva,b(|a|,|b|) such that

$$\mathrm{leva,b}(|\mathrm{a}|,|\mathrm{b}| =$$

$$\begin{cases} \max(i,j) & if\ min(i,j) = 0, \\ min \begin{cases} \mathrm{lev}_{\mathrm{a,b}}(\mathrm{i}-1,\mathrm{j})+1 \\ \mathrm{lev}_{\mathrm{a,b}}(\mathrm{i},\mathrm{j}-1)+1 \\ \mathrm{lev}_{\mathrm{a,b}}(\mathrm{i}-1,\mathrm{j}-1)+1_{(\mathrm{a}_{\mathrm{i}\neq \mathrm{b}_\mathrm{j}})} \end{cases} & otherwise^{\cdots} \end{cases}$$

....Eq.(2)

For example, the Levenshtein distance among the two strings sequences "kitten" and "sitting" is calculated as 3, here "kitten is the source string sequence (s) and "sitting" is target object string (t) "since the following three edits are necessary to convert string sequence (s) and the target object string (t), and there is no other way to accomplish it with less than three edits:

- kitten → sitten (substitution of 's' for 'k')
- sitten → sittin (substitution of 'i' for 'e')

- sittin → sitting (insertion of 'g' at the end). Underneath below is the Levenstein distance algorithm pseudo code:

Stage 1: Initialization

- Set k to be the extent of s, set l to be the extent of t.
- Construct a template matrix that enclosing 0……..k rows and 0………l columns.
- Starts with the original row to 0………..k.
- Starts with the column to 0…………….l.

Step2: Processing

- Examine s (i from 1 to k).
- Examine t (j from 1 to l).
- If s[i] is identical to t[j], then cost is 0.
- If s[i] doesn't identical t[j], then cost is 1.
- Set cell d[i,j] of the template matrix identical to the smallest of:
- The cell straight away above plus 1: d[i-1,j] + 1.
- The cell straight away to the left plus 1: d[i,j-1] + 1.
- The cell crossways above and to the left plus the cost: d[i-1,j-1] + cost.

Step 3: Output
Step 2 is continual till the d[k,l] value is found
Levenstein Distance [LD] has [74] a large diversity of functions such as spell checkers, modification systems for OCR and software means employed to facilitate out natural language conversion based on translation memory. The Levenshtein distance (LD) can also be employed as an support in finding fuzzy string similarity and searching in applications such as record linkage, the evaluated strings are characteristically short to make possible improve speed of estimation. Just like FAR-HD algorithm, EFAR-HD makes use of two phased practice, in the first stage the algorithm inspects each operation in the existing separation of the dataset and discover out the familiar candidate items, the task name build association do this job to discover out the familiar items in the dataset, we have the used the soybean dataset and construct a tidlist for each singleton originated. The levenstein distance (LD) here test out the dissimilarity of two strings sequences. Levenstein distance divides the dataset into two partitions. One separation contains related strings which are short in length and other separation contains the string sequences who having elongated length. The long length strings sequences are unnecessary and can't be further employed for rules pruning. According to levenstein distance (LD) logic, the string sequence whose calculated levenstein distance is more than three i.e. if a string sequence requires more than three edits is considered to be long length string which can't be used for rules pruning. After all singletons in the obtainable partition have been scrutinized, the Tidlists of singletons which are not d-familiar or whose calculated levenstein distance (LD) is more than three are dropped out.

The process of rules generation or pruning from the d-frequent items are done in the second stage of E-FARHD algorithm, during this stage , association rule mining (ARM) by using fuzzy has been made. The algorithm one by-one navigates each and every separation from the creation and discover out the familiar and recurrent candidate items over the entire dataset. The rules pruning are carried out on these frequent applicant items.

A. Pseudocode of EFARM algorithm [36]

Phase - I:

- Go across every severance st ∈ S do
- Go across every action a ∈ existing partition st do
- for all singleton s ∈ existing operation a d o
- Work out µ for every s
- If LD for si to s <1
- count[s]+=µ
- Else
- Swap syllables
- If LD sli to s<1
- counts[s]+=µ
- end If end for end for
- Go across every singleton si where i=1 to |Sp| do
- If si is not d-recurrent i.e (common candidate item) then exterminate Tid[si]
- end if
- end for
- Go across every singleton si where i=1 to |sp| do
- Go across every singleton sj where j=1 to |sp| do
- ProduceNewtItemSet(si,sj)
- end for end for end for

B. Pseudocode to create newitemset of commom candidate items

- ProduceNewItemSet:
- Come together IT and sf to get innovative item ITnew
- Tid[ITnew]=Tid[IT]∩Tid[ITsf]
- Work out µp for ITnew using tid[ITnew]
- count[ITnew]+=µp
- If IT new is familiar entrant item or d-recurrent then
- Go across every singleton sk where k=f+1 to |sp| do
- produceNewItemSet(ITnew,sk)
- end for end if
- Eradicate Tid[ITnew]

Phase 2

- Go across each separation st ∈ S do
- Go across each itemset IT ∈ st in the earliest stage do
- if IT is recurring in overindulgence of the entire dataset *E* then
- yield IT
- end if
- eradicate IT
- end for
- for all permanent itemset IT do
- organize constituent singletons $s1, s2, \dots, sn$ of IT such that $it = s_1 \cap s_2 \cap s_3 \cap \dots \dots \dots s_n$
- Tidlist is tid[IT] = interrelate Tidlists of apiece and every one crucial singletons

- work out μ for IT by employing Tid[IT]
- $count$[IT]+ = μ
- end for
- if no itemsets reside last to be count up then go out
- end if
- end for

EFAR-HD is designed and developed to work in proficient comportment, Just like the preceding algorithm EFAR-HD utilizes the same task and logic during the stage 2 and it is unmovable during the completion as this is the restructured only during the stage 1 of the preceding algorithm developed by Ashish Mangalampalli and Vikram Pudi. Moreover during stage 2 the algorithm calculates for each remaining itemset IT, determine its indispensable singletons $s1$, $s2$, st and then accomplish the Tidlist of IT(Tid[IT]) by interlocking the Tidlists of all the constituent singletons. Furthermore, the count up of all singleton IT is reorganized in [IT]. Thus, got switch over amongst outputting and deleting itemsets and generating Tidlists for itemsets in eagerness of no complementary itemsets are stay at the back.

IV. EXPERIMENTAL SETUP AND ANALYSIS

In this section, we give explanation about the experimental setup and analysis employed for contrasting EFAR-HD with two other Fuzzy ARM algorithms Fuzzy Apriori and FAR-HD – the detailed description about the first algorithm is described in [9] and [10] and the second one being above . We have restructured the FARHD and Fuzzy Apriori in the java programming language and additional the algorithm FAR-HD is improved by adding up the levenstein distance (LD) in the coding to check the extent of likeness or to work out the distance among two strings. Further we have established the conception of phonetics in the EFAR-HD which reinstates the character “ee” with the “i” if necessary to build the rule. The implementation of the algorithm is done on the eclipse kepler and connect it with the weka tool to find the associations between singular items. The weka tool offers the edge to hook up the dataset with the EFAR-HD algorithm.

V. EXPERIMENTAL RESULTS

Soybean Dataset: The aim of this soybean dataset (small) is to assist soybean disease diagnosis based on viewed morphological features. This dataset includes 47 instances with 35 categorical attributes out of which some are nominal while some are ordered attributes. The soybean dataset contain an attribute name “dna” that implies “does not mean”. The values of attributes are programmed numerically with the initial value encoded as” 0” and the subsequent as “1” and so on. An unknown value is marked or encoded as “?”. The whole dataset is categorized into four different classes of diseases viz., D1, D2, D3 and D4. The number of instance cases belongs to each class D1, D2, D3 and D4 are 10, 10, 10 and 17 respectively. This dataset is a 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.

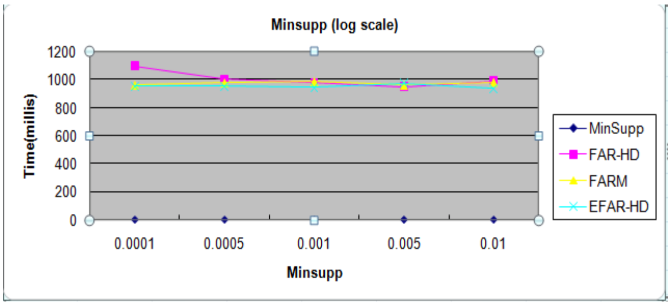

Fig 1. Minsupp Support from 0.0001 to 0.1

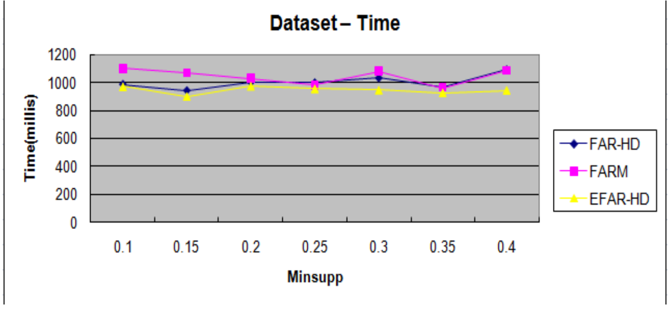

Fig 2. Minsupp Support from 0.1 to 0.4

A. Rules generated by FARHD in Milisec by WEKA Associator

Best 10 rules found:

- int-discolor=none 4648 ==> sclerotia=absent 4648
- mycelium=absent int-discolor=none 4600 ==> sclerotia=absent 4600
- leaves=abnorm sclerotia=absent 4384 ==> mycelium=absent 4376
- sclerotia=absent 5000 ==> mycelium=absent 4952
- int-discolor=none 4648 ==> mycelium=absent 4600
- int-discolor=none sclerotia=absent 4648 ==> mycelium=absent 4600
- int-discolor=none 4648 ==> mycelium=absent sclerotia=absent 4600
- leaf-malf=absent 4432 ==> mycelium=absent 4384
- mycelium=absent 5112 ==> sclerotia=absent 4952
- leaves=abnorm mycelium=absent 4536 ==> sclerotia=absent 4376

Time taken (in millis) 1005

B. Rules generated by E-FARHD in milisec in WEKA Associator

Best rules found:

- int-discolor=none 4648 ==> sclerotia=absent 4648
- mycelium=absent int-discolor=none 4600 ==> sclerotia=absent 4600
- leaves=abnorm sclerotia=absent 4384 ==> mycelium=absent 4376
- sclerotia=absent 5000 ==> mycelium=absent 4952
- int-discolor=none sclerotia=absent 4648 ==> mycelium=absent 4600
- leaf-malf=absent 4432 ==> mycelium=absent 4384
- mycelium=absent 5112 ==> sclerotia=absent 4952

- leaves=abnorm mycelium=absent 4536 ==> sclerotia=absent 4376

Time taken(in millis.) 975

C. Rules generated by FARM in milisec in WEKA Associator

- int-discolor=none 4648 ==> sclerotia=absent 4648
- mycelium=absent int-discolor=none 4600 ==> sclerotia=absent 4600
- leaves=abnorm sclerotia=absent 4384 ==> mycelium=absent 4376
- sclerotia=absent 5000 ==> mycelium=absent 4952
- int-discolor=none 4648 ==> mycelium=absent 4600
- int-discolor=none sclerotia=absent 4648 ==> mycelium=absent 4600
- int-discolor=none 4648 ==> mycelium=absent sclerotia=absent 4600
- leaf-malf=absent 4432 ==> mycelium=absent 4384
- mycelium=absent 5112 ==> sclerotia=absent 4952 conf:(0.97)
- leaves=abnorm mycelium=absent 4536 ==> sclerotia=absent 4376

Time taken (in millis.)1017

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 item sets and also contains the item sets 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. Weka associator reveal the outcome acquire by running EFAR-HD, FARM, and FAR-HD is that the EFAR-HD produce rules faster than FARM, and FAR-HD for minimum support values varying 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.

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

[9] A. 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] A. 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] [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 www.iosrjournals.org 8

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

[40] 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.

[41] 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.

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

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

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

[45] 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.

[46] 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.

[47] 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.

[48] 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.

[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.

[50] 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.

[51] T. P. Hong, et al., "Mining fuzzy multiple-level association rules from quantitative data," Applied Intelligence, Volume 18, Page(s): 79-90, Jan-Feb, 2003.

[52] 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.

[53] 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.

[54] 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.

[55] 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.

[56] 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.

[57] 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.

[58] 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.

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

[60] 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.

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

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

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

[64] 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.

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

[66] "Data Mining Curriculum". ACM SIGKDD. 2006-04-30. Retrieved 2011-10-28. Clifton, Christopher (2010). "Encyclopædia Britannica: Definition of Data Mining". Retrieved 2010-12-09.

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

[68] 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

[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, l996,

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

[71] 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.

[72] 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.

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