International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988 Volume 6 (2014) pp. 11 - 22

Decision Making For Items To Be Kept For Sale In Supermarket Using Fuzzy-Genetic Approach with Single Minimum Support Using 3-Dimensional k-means Clustering

Shaikh Nikhat Fatma[1] , Dr. J W Bakal[2]

[1] Department of Computer, Mumbai University,
Pillai Institute of Information Technology, Engineering, Media Studies & Research
New Panvel, India
nikhats10@yahoo.com

[2] Department of Information Technology, Mumbai University,
Mumbai, India
bakaljw@gmail.com

Abstract: **Data mining is the process of extracting desirable knowledge or interesting patterns from existing databases for specific purposes. Most conventional data-mining algorithms identify the relationships among transactions using binary values. Transactions with quantitative values are however commonly seen in real-world applications. The fuzzy concepts are used to represent item importance, item quantities, minimum supports and minimum confidences. Fuzzy operation like intersection is used to find large itemsets and association rules. Each attribute uses only the linguistic term with the maximum cardinality in the mining process. It uses a combination of large 1-itemsets and membership-function suitability to evaluate the fitness values of chromosomes. The calculation for large 1-itemsets could take a lot of time, especially when the database to be scanned could not totally fed into main memory. In this system, an enhanced approach, called the 3-dimensional k-means cluster-based fuzzy-genetic mining algorithm is used, which uses the coverage factor overlap factor and average of both factors to cluster the chromosomes. It divides the chromosomes in a population into clusters by the 3-dimensional k-means clustering approach and evaluates each individual according to both cluster and their own information. A genetic- fuzzy data-mining algorithm for extracting fit membership functions and multilevel association rules with its confidence from quantitative transactions is shown in this paper**.

Keywords: ***3 dimensional k-means Clustering, data mining, fuzzy set, Fuzzy Mining(FM), genetic algorithm, chromosomes, confidence, Fuzzy Association Rules, membership functions, Quantitative transactions***

I. Introduction

The information through data mining can be converted into reliable and business oriented trends and patterns; for instance a sales manager can use data mining to analyze his daily sales summary/information in several aspects like getting to know consumer behavior or may be target consumers. Hence he can increase his revenue of business by knowing such information. Similarly, data mining helps the companies to identify their profitable elements like at what price product should be sold? what should be the product positioning?, finding economic indicators, customer behavior and demographics. All these elements and factors facilitate the companies in finding their competitive advantage. Data is very important for every supermarket. Data that was measured in gigabytes until recently, is now being measured in terabytes, and will soon approach the pentabyte range. In order to achieve our goals, we need to fully exploit this data by extracting all the useful information from it. This information can be extracted using data mining. Data mining is the use of automated data analysis techniques to uncover previously undetected relationships among data items. Data mining often involves the analysis of data stored in large database. The typical business decision that the management of a super market has to make is what item to put for sale and in what quantity. Analysis of past transaction data is a commonly used approach in order to improve the quality of such decisions. Extraction of frequent item sets is essential towards mining interesting patterns from datasets. A typical usage scenario for searching frequent patterns is the so called "market basket analysis" that involves analyzing the transactional data of a super market in order to determine which products are purchased together and how often and also examine customer purchase preferences.

Mining association rules between sets of items in large databases was first stated by Agrawal, Imelinski and Swami in 1993 and it opened brand new family algorithms .Apriori algorithm is probably the most used algorithm in association rules mining. At present, more and more databases

containing large quantities of data are available. These industrial, medical, financial and other databases make an invaluable resource of useful knowledge. The task of extraction of useful knowledge from databases is challenged by the techniques called data-mining techniques. One of the widely used data-mining techniques is association rules mining. Data Mining is commonly used in attempts to induce association rules from transaction data. Data mining is the process of extracting desirable knowledge or interesting patterns from existing databases for specific purposes. Many types of knowledge and technology have been proposed for data mining. Among them, finding association rules from transaction data is most commonly seen. Most studies have shown how binary valued transaction data may be handled. Transaction data in real-world applications, however, usually consist of fuzzy and quantitative values.[1][2]

In fuzzy data mining, the first thing that it needs to be done is to define appropriate membership functions because they have a critical influence on the final mining results. Items may have different importance, which is evaluated by managers or experts as linguistic terms. Membership functions are defined by experts. That is the best approach, absolutely. However, experts may not always do this since the customers' favorites change all the time. Most of the previous fuzzy data mining algorithms thus assume the membership functions are already known. The algorithms that can derive both the appropriate membership functions and fuzzy rules automatically are thus developed. A survey of several algorithms that can mine both appropriate membership functions and fuzzy association rules were made. It can be divided into four different genetic-fuzzy data mining problems according to the utilized approaches and two types of problems in fuzzy data mining, namely Integrated Genetic-Fuzzy approach for items with Single Minimum Supports (IGFSMS) Integrated Genetic-Fuzzy approaches for items with Multiple Minimum Supports (IGFMMS), Divide-and-Conquer Genetic-Fuzzy approaches for items with Single Minimum Supports (DGFSMS) and Divide-and-Conquer Genetic-Fuzzy approaches for items with Multiple Minimum Supports (DGFMMS) problems as shown in Table 1. In the Table 1, in the integrated genetic-fuzzy approaches, they encoded all membership functions of all items (attributes) into a chromosome (also called an individual). The genetic algorithms are then used to derive a set of appropriate membership functions according to the designed fitness function. Finally, the best set of membership functions are then used to mine fuzzy association rules. On the other hand, the divide-and-conquer genetic-fuzzy approaches go in different direction. They encoded membership functions of each item into a chromosome. In other words, chromosomes in a population were maintained just for only one item.[16]

*Table 1.*The four different genetic-fuzzy data mining problems

	INTEGRATED APPROACH	**DIVIDE AND CONQUER APPROACH**
SINGLE MINIMUM SUPPORT	IGFSMS Problem	IGFSMS Problem
MULTIPLE MINIMUM SUPPORT	IGFMMS Problem	DGFMMS Problem

II. Association Rule Mining

Agrawal and his co-workers proposed several mining algorithms based on the concept of large itemsets to find association rules in transaction data. They divided the mining process into two phases. In the first phase, candidate itemsets were generated and counted by scanning the transaction data. If the number of an itemset appearing in the transactions was larger than a pre-defined threshold value (called minimum support), the itemset was considered a large itemset. Itemsets containing only one item were processed first. Large itemsets containing only single items were then combined to form candidate itemsets containing two items. This process was repeated until all large itemsets had been found. In the second phase, association rules were induced from the large itemsets found in the first phase. All possible association combinations for each large itemset were formed, and those with calculated confidence values larger than a predefined threshold (called minimum confidence) were output as association rules.[15]

For example, huge amounts of customer purchase data are collected daily at the checkout counters of grocery stores. Table 2 illustrates an example of such data, commonly known as market basket transactions. Each row in this table corresponds to a transaction, which contains a unique identifier labeled TID and a set of items bought by a given customer. Retailers are interested in analyzing the data to learn about the purchasing behavior of their customers. Such valuable information can be used to support a variety of business-related applications such as marketing promotions, inventory management, and customer relationship management.

This methodology is known as association analysis, which is useful for discovering interesting relationships hidden in large data sets. The uncovered relationships can be represented in the form of association rules or sets of frequent items. For example, the following rule can be extracted from the data set shown in Table 1:

{Diapers} → {Beer}.

Table 2. An example of market basket transactions.

TID	**Items**
1	{Bread, Milk}
2	{Bread, Diapers, Beer, Eggs}
3	{Milk, Diapers, Beer, Cola}
4	{Bread, Milk, Diapers, Beer}
5	{Bread, Milk, Diapers, Cola}

The rule suggests that a strong relationship exists between the sale of diapers and beer because many customers who buy diapers also buy beer. Retailers can use this type of rules to help them identify new opportunities for cross selling their products to the customers.

The major steps in association rule mining are:

1. Frequent Item sets generation
2. Rules derivation

III. Fuzzy Set Concepts

Fuzzy set theory was first proposed by Zadeh and Goguen in 1965. A function called the membership function, $\mu A(x)$, is defined for mapping a member x to a membership degree between 0 to 1. Triangular membership functions are commonly used and can be denoted by $A = (a, b, c)$, where $a \leq b \leq c$. The abscissa b represents the variable value with the maximal grade of membership value, i.e. $\mu A(b)=1$; a and c are the lower and upper bounds of the available area. They are used to reflect the fuzziness of the data. the fuzzy concepts are used to represent item importance, minimum supports and minimum confidences. These parameters are expressed in linguistic terms, which are more natural and understandable for human beings.

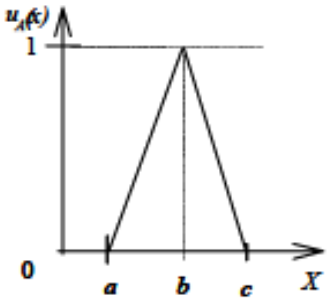


Figure 1. A triangular membership function

In this paper, the fuzzy concepts are used to represent item importance, minimum supports and minimum confidences. These parameters are expressed in linguistic terms, which are more natural and understandable for human beings. There are a variety of fuzzy set operations. Among them, three basic and commonly used operations are complement, union and intersection[15]. In this paper we have used intersection operation.

IV. Fuzzy Mining

For many applications, an association rule may be more interesting if it reveals relationship among some useful concepts, such as "high income", "new car", and "frequent customer". These concepts are often imprecise or uncertain. Interesting concepts are defined using fuzzy terms and interpreted based on fuzzy set. We refer association rules involving fuzzy terms as *fuzzy quantitative association rules*. A fuzzy data mining algorithm is specially capable of transforming quantitative values in transactions into linguistic terms, then filtering them, and finding association rules by modifying the apriori mining algorithm. For example, Age = young and income = high →Risk Level = medium High is a fuzzy quantitative association rule, where "young", "high" and "medium High" are fuzzy terms. This is a process of fuzzifying numerical numbers into linguistic terms, which is often used to reduce information overload in hurnan decision making process. The numerical salary, for example, may be perceived in linguistic terms as high, average and low. One way of determining membership functions of these linguistic terms is by expert opinion or by people's perception. Fuzzy association rules use linguistic variables. It has linguistic inputs and outputs, which are more natural and understandable for human beings. These linguistic variables define the value of a variable qualitatively, by defining a symbol for a fuzzy set, and by defining the meaning of the fuzzy set. Basically, the fuzzy mining algorithms first uses membership functions to transform each quantitative value into a fuzzy set in linguistic terms. The algorithm then calculated the scalar cardinality of each linguistic term on all the transaction data. The mining process based on fuzzy counts was then performed to find fuzzy association rules.

Table 3: An example of fuzzy dataset.

ID	Age	Degree	Salary
E1	Adult	M.Tech.	30000 (High)
E2	Old	B.A.	18000 (Normal)
E3	Young	B.Tech.	28000 (High)
E4	Adult	M.C.A.	10000 (Low)

Table 2 contains a sample fuzzy dataset. We can determine the value of the attribute i_k of the j^{th} record by using the convention $t_j[i_k]$. For example, if we want to determine the value of salary of third record, we will write t_3[Salary] and obtain the value 28000. In Table 1, the attribute salary has been denoted using the fuzzy set Salary = {high, normal, low}, dividing the salary interval into low, normal and high. For the interval (Rs. 10000 to Rs. 30000) we have normal salary, for (Rs. 10,000 and below) we have low salary and for (Rs.30,000 and above)we have high salary.

Basically, the fuzzy mining (FM) algorithms first uses membership functions to transform each quantitative value into a fuzzy set in linguistic terms. The algorithm then calculated the scalar cardinality of each linguistic term on all the transaction data. The mining process based on fuzzy counts is then performed to find fuzzy association rules.

V. Drawbacks of Fuzzy Mining

In fuzzy data mining, the first thing that it needs to be done is to define appropriate membership functions because they have a critical influence on the final mining results. Membership functions are defined by experts. That is the best approach, absolutely. However, experts may not always do this since the customers' favorites change all the time. Most of the previous fuzzy data mining algorithms thus assume the membership functions are already known. Of course, it is not

suitable when we try to apply it to real applications. The algorithms that can derive both the appropriate membership functions and fuzzy rules automatically are thus developed. Most of these fuzzy data mining algorithms assume that the membership functions are already known. The developing of mining algorithms that can mine both appropriate membership functions and fuzzy association rules automatically is thus an important task. There are two reasons for the drawback of fuzzy mining. The first one is that companies may not always ask experts to define the appropriate membership function because it needs to spend lots of money and time. The second reason is that the favorite things of customers change all the time. Some mechanisms are thus needed to adapt the membership functions to these changes automatically. This can be done by the Genetic Algorithm.[16]

VI. Genetic Algorithm Concept

Genetic Algorithms were first developed by computer scientist John Holland in the 1970's as an experiment to see if computer programs could evolve in the Darwinian sense. It is a based on the theory of natural selection in that it takes a population of 'solutions' to a problem, and uses them to 'breed' solutions that take the best 'genes' or characteristics of their parents. Instead of the ability to survive, parent solutions are allowed to mate if they are the best in the population at solving the problem. Since the microcomputer can cycle through a generation in a split second, millions of generations of good breeding can be compacted into a short period of time, and the best offspring can be chosen as the solution to the problem.

A. Chromosome: A set of genes. Chromosome contains the solution in form of genes.

B. *Gene:* A part of chromosome. A gene contains a part of solution. It determines the solution. E.g. 16743 is a chromosome and 1, 6, 7, 4 and 3 are its genes.

C. Individual: Same as chromosome.

D. *Population:* No of individuals present with same length of chromosome.

E. *Fitness:* Fitness is the value assigned to an individual. It is based on how far or close a individual is from the solution. Greater the fitness value better the solution it contains.

F. Fitness function: Fitness function is a function which assigns fitness value to the individual. It is problem specific.

G. Breeding: Taking two fit individuals and intermingling there chromosome to create new two individuals.

H. Mutation: Changing a random gene in an individual.

I. Selection: Selecting individuals for creating the next generation.

VII. Existing System

In the past, a fuzzy-genetic data-mining algorithm for extracting both association rules and membership functions from quantitative transactions. It used a combination of large 1-itemsets and membership-function suitability to evaluate the fitness values of chromosomes. The calculation for large 1-itemsets could take a lot of time, especially when the database to be scanned could not totally fed into main memory.

Then fuzzy, GA and clustering concepts are used to discover both useful fuzzy association rules and suitable membership functions from quantitative transactions. A cluster-based fuzzy-GA mining framework first used for searching membership functions suitable for the mining problem and then using the final best set of membership functions to mine fuzzy association rules. [1][2]

The existing framework will maintain a population of sets of membership functions, and use the genetic algorithm to automatically derive the resulting one. It will first transform each set of membership functions into a fixed-length string.

Each chromosome will represent a set of membership functions used in fuzzy mining. Then, it will use the k-means clustering approach to gather similar chromosomes into groups. All the chromosomes in a cluster will use the number of large 1-itemsets derived from the representative chromosome in the cluster and their own suitability of membership functions to calculate their fitness values. Since the number for scanning a database will decrease, the evaluation cost can thus be reduced .The evaluation results can be utilized to choose appropriate chromosomes for mating in the next generation. The offspring membership function sets will then undergo recursive “evolution” until a good set of membership functions has been obtained. Finally, the derived membership functions will be used to mine fuzzy association rules.[1][2]

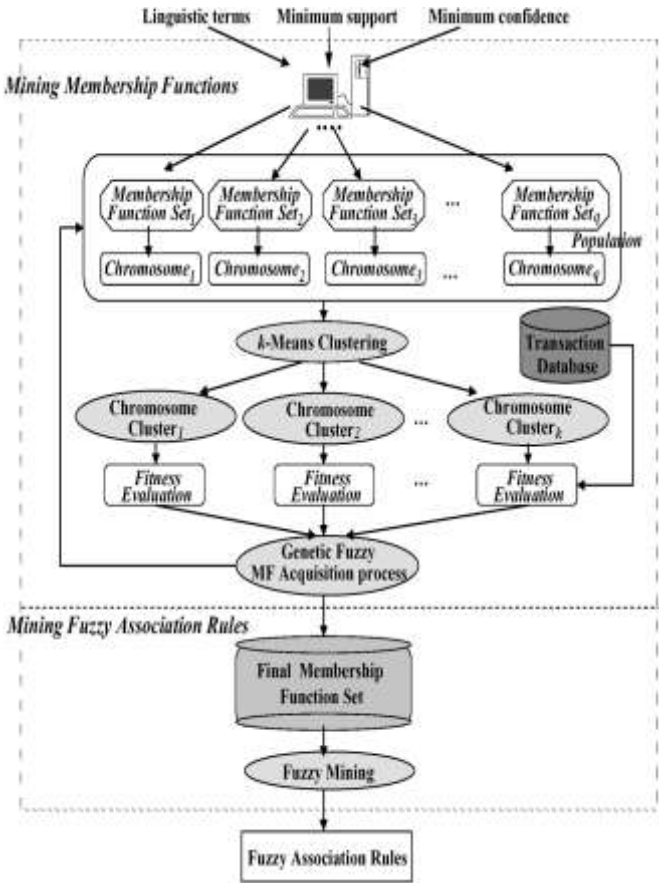


Figure 2. Framework for the existing cluster based fuzzy genetic data mining system

VIII. Proposed System

The proposed framework maintains a population of sets of membership functions , and uses the genetic algorithm to automatically derive the resulting one. It first transforms each set of membership functions into a fixed-length string. It then chooses appropriate strings for "mating", gradually creating good offspring membership function sets. The offspring membership function sets then undergo recursive "evolution" until a good set of membership functions has been obtained.

A. Chromosome Representation

It is important to encode membership functions as string representation for GA to be applied. Here, each set of membership functions is encoded as shown below. In Figure 3, each membership function is assumed to be isosceles-triangle and represented by a pair(c,w) ,with c indicating the centre abscissa and w representing half the spread. R_{jk} denotes the membership function of the k^{th} linguistic term of *item* I_j, , c_{jk} indicates the center abscissa of fuzzy region R_{jk}, and w_{jk} represents half the spread of fuzzy region R_{jk} . All pairs of (c, w)'s for a certain item are concatenated to represent its membership functions. The set of membership functions MF_1
for the first item I_1 is then represented as a substring of $c_{11}w_{11}$ . . . $c_{1|I_1|}w_{1|I_1|}$, where $|I_1|$ is the number of terms of I_1. The entire set of membership functions is then encoded by concatenating substrings of MF_1,MF_2, ...,MF_j .Since c and w are both numeric values, a chromosome is thus encoded as a fixed-length real-number string rather than a bit string. Other types of membership functions (e.g non-isosceles trapezes) can also be adopted in our method. For coding non-isosceles triangles and trapezes, three and four points are needed instead of two for isosceles triangles. Besides, the number of membership function for each item can be different. In our system we assume each item has three linguistic terms, low, middle and high for the membership function. The number of linguistic terms may have impact on the results. However, how to decide the appropriate number of membership functions is a complex problem.[1][2]

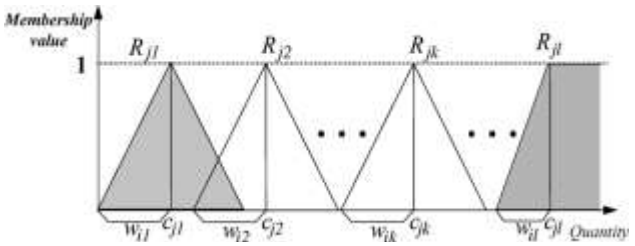


Figure 3. Membership Function of Item I_j

A. Initial Population

A genetic algorithm requires a population of feasible solutions to be initialized and updated during the evolution process. As mentioned above, each individual within the population is a set of isosceles-triangular membership functions. Each membership function corresponds to a linguistic term in a certain item. The initial set of chromosomes is randomly generated with some constraints for forming feasible membership functions.[1][2]

B. Fitness and Selection

In order to develop a good set of membership functions from an initial population, the genetic algorithm selects *parent* membership function sets with its probability values for mating. An evaluation function is then used to qualify the derived membership function sets. The performance of membership function sets is then fed back to the genetic algorithm to control how the solution space is searched to promote the quality of the membership functions. [1][2]

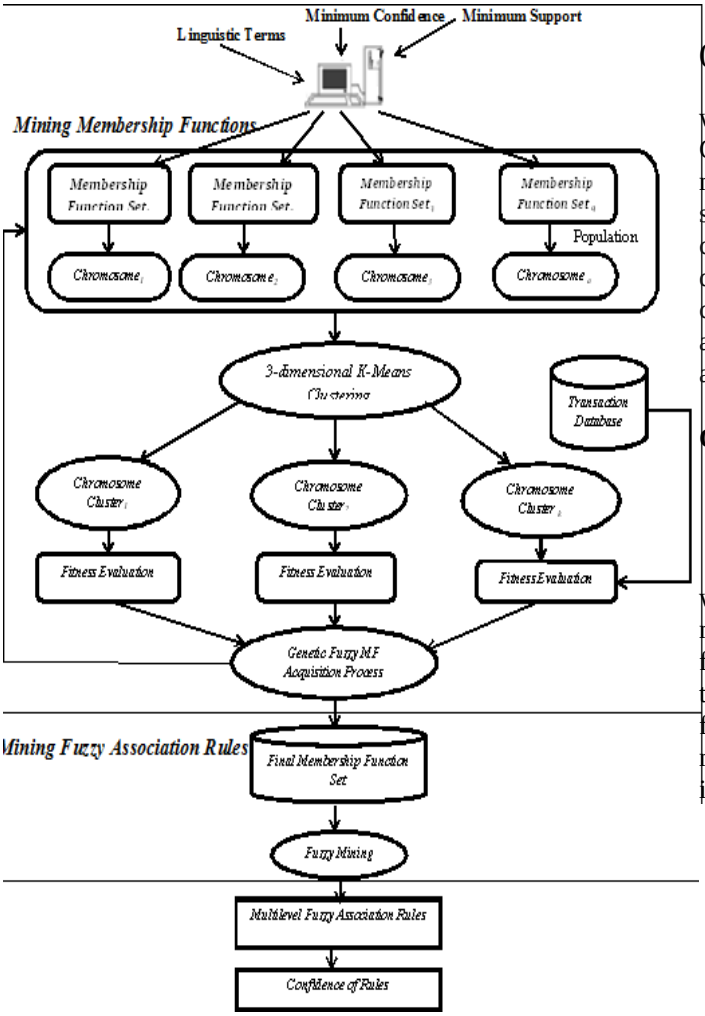


Figure 4. Framework for the proposed 3- dimensional cluster based fuzzy genetic data mining system

$$f(C_q) = \frac{|L_{1q}|}{\text{Suitability}(C_q)} \quad (1)$$

Where $|L1q|$ is the number of large 1-itemsets obtained by using the set of membership functions in chromosome Cq and Suitability (Cq) represents the shape suitability of Cq. Suitability (Cq) is defined as

$$\sum_{j=1}^{m} [\text{overlap\_factor}(C_{qj}) + \text{coverage\_factor}(C_{qj})] \quad (2)$$

where m is the number of items. Overlap_factor (C_{qj}) represents the overlap factor of the membership functions for an item I_j in the chromosome C_q The overlap ratio of two membership functions R_{jk} and R_{ji} is defined as the overlap length divided by half the minimum span of the two functions. If the overlap length is larger than half the span, then these two membership functions are thought of as a little redundant. Thus, the overlap factor of the membership functions for an item I_j in the chromosome C_q is defined as

$$\text{overlap\_factor}(C_{qj}) = \sum_{k \neq i} [\max((\frac{\text{overlap}(R_{jk}, R_{ji})}{\min(w_{jk}, w_{ji})}), 1) - 1] \quad (3)$$

where overlap (R_{jk}, R_{ji}) is the overlap length of R_{jk} and R_{ji}. Coverage Factor (C_{qj}) represents the coverage ratio of a set of membership functions for an item I_j The coverage ratio of a set of membership functions for an item I_j is defined as the coverage range of the functions divided by the maximum quantity of that item in the transactions. The more the coverage ratio is, the better the derived membership functions are. Thus, the coverage factor of the membership functions for an item I_j in the chromosome C_q is defined as

$$\text{Coverage\_factor}(C_{qj}) = \frac{1}{\frac{\text{Range}(R_{j1},\ldots..,R_{jl})}{\max(I_j)}} \quad (4)$$

where range (R_{j1},R_{j2},…..,R_{jl}) is the coverage range of the membership functions, l is the number of membership functions for I_j , and max (I_j) is the maximum quantity of I_j in the transactions. The suitability factor used in the fitness function can reduce the occurrence of the two bad kinds of membership functions shown in Figure. 5, where the first one is too redundant, and the second one is too separate.[1][2]

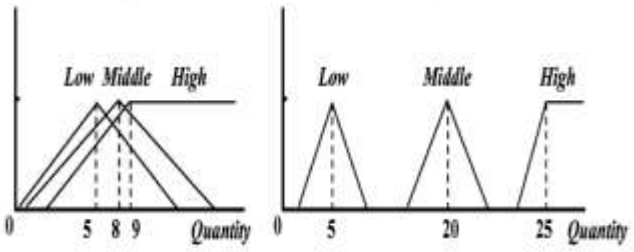


Figure 5. Two Bad Membership Functions

The overlap factor in suitable(C_q) is designed for avoiding the first bad case, and the coverage factor is for the second one.

C. Clustering Chromosomes

Although the evaluation by 1-itemsets is much faster than that by all itemsets or interesting association rules, it is still time-consuming since the database must be scanned once for each chromosome. We thus propose a new method based on clustering technique to reduce the evaluation time of large 1-itemsets. The process proceeds as follows. The coverage factors and overlap factors of all the chromosomes are used to form appropriate clusters. The 3-dimensional k -means clustering approach is adopted here to cluster chromosomes. Since the chromosomes with similar coverage factors (CF) and overlap factors (OF) will form a cluster, they will have nearly the same shape of membership functions and induce about the same number of large 1-itemsets. For each cluster, the chromosome which is the nearest to the cluster center is thus chosen to derive its number of large 1-itemsets. All chromosomes in the same cluster then use the number of large 1-itemsets derived from the representative chromosome as their own. Finally, each chromosome is evaluated by this number of large 1-itemsets divided by its own suitability value.

D. Genetic Operators

Genetic operators are very important to the success of specific GA applications. Two genetic operators, the *crossover* and the *mutation*, are used in the genetic-fuzzy mining framework. Assume there are two parent chromosomes

$\mathbf{C_u^t = (c_1,\ldots\ldots,c_h,\ldots\ldots,c_z)}$, **and**

$\mathbf{C_w^t = (c_1',\ldots\ldots, c_h',\ldots\ldots, c_z')}$. **(5)**

The crossover operator will generate the following four candidate chromosomes from them:

1) $\mathbf{C_1^{t+1} = (c_{11}^{t+1},\ldots, c_{1h}^{t+1},\ldots, c_{1z}^{t+1})}$, **where** $\mathbf{c_{1h}^{t+1} = d_{ch} + (1\text{-}d)\, c_h'}$ **;**

2) $\mathbf{C_2^{t+1} = (c_{21}^{t+1},\ldots, c_{2h}^{t+1},\ldots, c_{2z}^{t+1})}$, **where** $\mathbf{c_{2h}^{t+1} = d_{c'h} + (1\text{-}d)\, c_h}$ **;**

3) $\mathbf{C_3^{t+1} = (c_{31}^{t+1},\ldots, c_{3h}^{t+1},\ldots, c_{3z}^{t+1})}$, **where** $\mathbf{c_{3h}^{t+1} = \min\{c_h, c_h'\}}$**;**

4) $\mathbf{C_4^{t+1} = (c_{41}^{t+1},\ldots, c_{4h}^{t+1},\ldots, c_{4z}^{t+1})}$, **where** $\mathbf{c_{4h}^{t+1} = \max\{c_h, c_h'\}}$**;** **(6)**

IX. Algorithm

Notation used in the paper are stated as follows.
N: the total number of transaction data;
m: the total number of attributes;
A_j : the j^{th} attribute,1<=j<=m;
$|A_j|$: the number of fuzzy regions for Aj;
R_{jk} : the k^{th} fuzzy region of A_j , $1 \le k \le |A_j|$;
$D^{(i)}$: the i^{th} transaction datum, $1 \le i \le n$;
$V_j^{(i)}$: the quantitative value Of A_j for $D^{(i)}$
$F_j^{(i)}$: the fuzzy set converted from $V_j^{(i)}$
$f_{jk}^{(i)}$: the membership value of v in Region R_{Jk} ;
$count_{jk}$: the summation of f_{jk} for i=1 to n;
α : the predefined minimum support level;
λ : the predefined minimum confidence value;
C_r. the set of candidate itemsets with r attributes (items);
L_{rr} : the set of large itemsets with r attributes (items).[4]

INPUT

1. A body of n quantitative transactions,
2. A set of m items, each with a number of linguistic terms.
3. A membership function for each item is to be given.
4. A parameter k for 3-dimensional k-means clustering
5. A population size P
6. A crossover point Pc
7. A mutation point Pm
8. A support threshold α
9. A confidence threshold λ
10. A fitness threshold

OUTPUT

Multilevel Fuzzy Association Rule with Confidence .

To get a set of fit membership functions by using Genetic Algorithm and 3-dimensional k-means Algorithm Approach.

1. Randomly generate a population of P individuals, each individual is a set of membership functions for all the m items.
2. Encode each set of membership functions into a string representation.
3. Calculate the coverage factor (CF) and the overlap factor (OF) according to the formula mentioned above and average of both coverage and overlap factor of each chromosome.
4. Divide the chromosomes into k clusters by using the 3-dimensional k-means clustering based on three attributes i.e coverage factor, overlap factor and average of both.
5. Find out the representative chromosome in each cluster, which is the nearest to the centre.
6. Calculate the number of large 1-itemsets for each representative chromosome by the following sub steps.

a) For each transaction data (items in the transaction table) D_i ,i=1 to n and for each item I_j , j=1 to m transform the quantitative value $V_j^{(i)}$ into a fuzzy set $f_j^{(i)}$ represented as *:*

$$\left(\frac{f_{j1}^{(i)}}{R_{j1}} + \frac{f_{j2}^{(i)}}{R_{j2}} + \cdots + \frac{f_{jl}^{(i)}}{R_{jl}}\right) \quad \textbf{(7)}$$

Using the above membership function representation by the chromosome where R_{jk} is the k^{th} fuzzy region (low, middle, high) of item I_j.
$f_{jl}^{(i)}$ is $V_j^{(i)}$'s fuzzy membership value in region R_{jk} and l(=$|A_j|$) is the number of linguistic terms for item I_j

b) For each item region R_{jk} , calculate it's scalar cardinality (count) on the transaction as follows:

$$\mathbf{count_{jk} = \sum_{i=1}^{n} f_{jk}^{(i)}} \quad \textbf{(8)}$$

c) For each $R_{jk,}$ $1 \le j \le m$ and $1 \le k \le |A_j|$, chech whether it's $count_{jk}$ is larger than or equal to the minimum support threshold α. If R_{jk} satisfies the above condition then it is put in the set of large 1-itemsets(L_1).

d) Set $|L_1|$ as the number of Large 1- itemsets for the representative chromosomes.

7. Calculate the fitness value of each chromosome using the number of large 1- itemsets of it's representative chromosomes.

$$\mathbf{f(C_q) = \frac{|L_{1q}|}{Suitability\ (C_q)}} \quad \textbf{(9)}$$

where suitability is :

$\mathbf{m}$

$$\sum_{j=1} [\text{overlap\_factor}(C_{qj}) + \text{Coverage Factor}(C_{qj})] \quad (10)$$

where m is the number of items.

8. Execute crossover operation on population..
9. Execute mutation operation on population.
10. Use the roulette wheel selection operation to choose appropriate individuals for the next generation.
11. If the termination criterion is not satisfied goto step 7; otherwise do the next step.
12. Get the set of membership functions with the highest fitness value.
13. This set of membership functions are then used to mine fuzzy association rules from the given transaction table.

To mine fuzzy association rules from the above membership functions the modified apriori algorithm is used.

1. Transform the quantitative values of each transaction, for each attribute into a fuzzy set using above found membership functions.
2. Calculate the count of each attribute region (linguistic term) R_{jk} in the transaction data.
3. Collect each attribute region (linguistic term) to form the candidate set C_1.
4. Check whether $count_{jk}$ of each R_{jk} is larger than or equal to the predefined minimum support value α . If R_{jk} satisfies the above condition, put it in the set of large 1-itemsets (L_1).That is:

$$L_1 = \{R_{jk} \mid count_{jk} \geq \alpha,\ 1 \leq j \leq m \text{ and } 1 \leq k \leq |A_j|\} \quad (11)$$

5. IF L_1 is not null, then do the next step; otherwise, exit the algorithm.
6. Set r=l, where *r* is used to represent the number of items kept in the current large itemsets.
7. Join the large itemsets L_r to generate the candidate set C_{r+1} in a way similar to that in the apriori algorithm, except that two regions (linguistic terms) belonging to the same attribute cannot simultaneously exist in an itemset in C_{r+1}. The algorithm first joins L_r and L_r under the condition that r-1 items in the two itemsets are the same and the other one is different. It then keeps in C_{r+1} the itemsets which have all their sub-itemsets of *r* items existing in L_r and do not have any two items R_{jp} and R_{jq} $(p \neq q)$ of the same attribute R_j.
8. Do the following substeps for each newly formed r+1 candidate itemset *s* with $(s_1, s_2, s_{3\ldots\ldots}, s_{r+1})$ *in* C_{r+1}:

(a) Calculate the fuzzy membership value of each transaction datum $D^{(i)}$. Here, the minimum operator is used for the intersection.

$$f_s^{(i)} = f_{s1}^{(i)} \wedge f_{s2}^{(i)} \wedge \ldots\ldots\ldots \wedge f_{sr+1}^{(i)} \quad (12)$$

where $f_{sj}^{(i)}$ is the membership value of $D^{(i)}$ in region s_j . If the minimum operator is used for the minimum operator is used for the intersection, then

$$f_s^{(i)} = \min_{j=1}^{r+1} f_{sj}^{(i)} \quad (13)$$

(b) Calculate the scalar cardinality (count) of each candidate 2-itemset in the transaction data as.

$$count_s = \sum_{i=1}^{n} f_s^{(i)} \quad (14)$$

(c) Check whether these counts are larger than or equal to the predefined minimum support value α, put *s* in Lr+1.

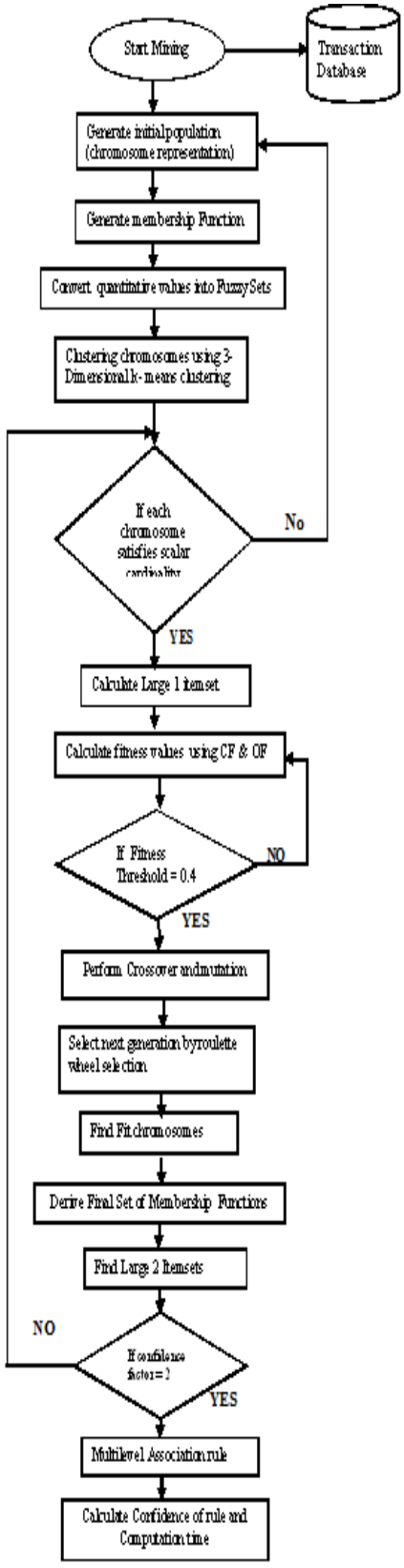


Figure 6. Flowchart for the proposed system

9. If L_{r+1} is null, then do the next step; otherwise, set r=r+l and repeat Steps 6 to 8.
10. Collect the large itemsets together.
11. Construct association rules for each large q-itemset *s* with items(s1, s2, ., sq), $q \geq 2$, using the following substeps:

(a) Form each possible association rule as follows:

$$s_1 \wedge \ldots \wedge s_{k-1} \wedge s_{k+1} \wedge \ldots \wedge s_q \rightarrow s_k \text{ , } k = 1 \text{ to } q. \quad (15)$$

(b) Calculate the confidence factors for the above association rules .

$$\frac{\sum_{i=1}^{n} f_s^{(i)}}{\sum_{i=1}^{n} (f_{s1}^{(i)} \wedge \ldots \wedge f_{sk-1}^{(i)}, f_{sk+1}^{(i)} \wedge \ldots \wedge f_{sq}^{(i)})} \quad (16)$$

12. Output multilevel association rules with confidence values larger than or equal to the predefined confidence threshold λ.

X. Experimental Results

The performance of the proposed approach is described. It was implemented in Java (Netbeans 6.1) on a personal computer with Intel Core i7, 2.00 GHz and 512MB RAM. A total of 10 items and 20 transactions were used in the experiments. In each data set, the numbers of purchased items in transactions were first randomly generated. The purchased items and their quantities in each transaction were then generated. An item could not be generated twice in a transaction. The quantitative transactions are 20 with 10 items and 3 linguistic terms ,low, Middle and High . The parameter k for 3-dimensional k-means clustering is 3. The initial population size P is set at 10, the crossover rate p_c is set at 6, and the mutation is set as random exchange of 2 points in same chromosome after crossover ,the minimum support α is set at 2.0, the fitness threshold is 2.0 and the confidence threshold λ is set at 2.0 .

A. Randomly Generated Chromosomes

Assume there are ten items in a transaction database: milk, bread, cookies , beverage, chocolate , icecream ,coldrink, curd fruit ,butter . Since the item *milk* has three possible linguistic terms, *Low, Middle and High,* the membership functions for *milk* are thus encoded as (2, 10,12,10, 22,10) for chromosome C2.

C0: 0 10 10 10 20 10, 4 7 11 7 18 7, 0 8 8 8 16 8, 0 7 7 7 14 7, 2 5 7 5 12 5, 0 2 2 2 4 2, 0 3 3 3 6 3, 0 3 3 3 6 3, 0 2 2 2 4 2, 0 6 6 6 12 6

C1: 0 6 6 6 12 6, 4 6 10 6 16 6 ,3 8 11 8 19 8, 0 9 9 9 18 9, 0 6 6 6 12 6, 0 2 2 2 4 2, 1 7 8 7 15 7, 0 4 4 4 8 4, 1 2 3 2 5 2, 0 2 2 2 4 2

C2: 2 10 12 10 22 10, 2 6 8 6 14 6, 3 5 8 5 13 5, 2 8 10 8 18 8, 0 6 6 6 12 6, 1 2 3 2 5 2, 1 7 8 7 15 7, 0 3 3 3 6 3, 1 2 3 2 5 2 ,1 3 4 3 7 3

C3: 0 6 6 6 12 6, 2 7 9 7 16 7, 1 8 9 8 17 8, 0 6 6 6 12 6, 0 3 3 3 6 3, 1 3 4 3 7 3, 0 4 4 4 8 4, 2 5 7 5 12 5, 1 3 4 3 7 3, 1 7 8 7 15 7

C4: 1 11 12 11 23 11, 3 13 16 13 29 13, 2 10 12 10 22 10, 2 3 5 3 8 3, 0 4 4 4 8 4, 0 2 2 2 4 2, 2 3 5 3 8 3, 0 6 6 6 12 6, 1 3 4 3 7 3, 0 8 8 8 16 8

C5: 0 6 6 6 12 6, 0 8 8 8 16 8, 1 9 10 9 19 9, 1 9 10 9 19 9, 2 6 8 6 14 6, 0 3 3 3 6 3, 2 6 8 6 14 6, 0 6 6 6 12 6, 0 4 4 4 8 4, 0 2 2 2 4 2

C6: 0 5 5 5 10 5, 0 9 9 9 18 9, 3 7 10 7 17 7, 2 8 10 8 18 8, 0 5 5 5 10 5, 0 2 2 2 4 2, 2 3 5 3 8 3, 2 4 6 4 10 4, 1 4 5 4 9 4 ,0 7 7 7 14 7

C7: 2 7 9 7 16 7, 4 13 17 13 30 13, 2 7 9 7 16 7, 0 8 8 8 16 8, 0 3 3 3 6 3, 0 2 2 2 4 2, 2 6 8 6 14 6, 0 3 3 3 6 3, 0 3 3 3 6 3, 1 2 3 2 5 2

C8: 2 11 13 11 24 11, 2 5 7 5 12 5, 1 8 9 8 17 8, 1 5 6 5 11 5, 1 6 7 6 13 6 ,1 2 3 2 5 2, 0 7 7 7 14 7, 1 5 6 5 11 5, 1 2 3 2 5 2, 0 6 6 6 12 6

C9: 1 3 4 3 7 3, 3 13 16 13 29 13, 2 7 9 7 16 7, 1 11 12 11 23 11, 1 4 5 4 9 4, 1 3 4 3 7 3, 0 4 4 4 8 4, 0 3 3 3 6 3, 0 4 4 4 8 4, 1 3 4 3 7 3

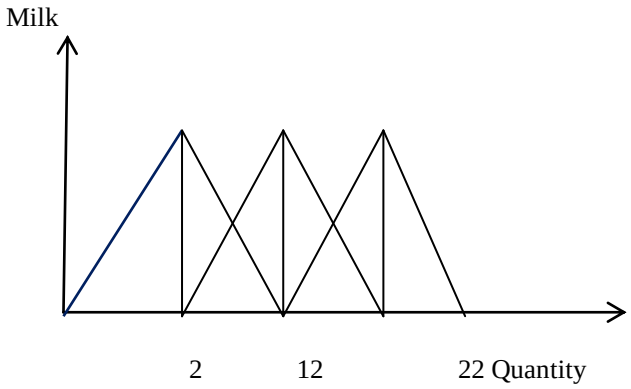


Figure 7. The membership functions for milk in C2

B. Calculation of overlap factor, coverage factor and both

The overlap factor in suitable(C_q) is designed for avoiding the first bad case, and the coverage factor is for the second one.[2] The dimensions used for clustering are overlap factor , coverage factor and average of both. Therefore, the calculation of minimum distance is as follows:

$$\sqrt{(x1-x2)2+(y1-y2)2+(z1-z2)2} \quad \textbf{(17)}$$

Table 4. Calculation of CF & OF & Average

Chromosome	Overlap Factor	Coverage Factor	Average of OF & CF
C0	0.0	0.79327214	0.4392281
C1	0.79327214	0.0	0.35404402
C2	0.4392281	0.35404402	0.0
C3	0.76132756	1.5545996	1.2005557
C4	0.7340702	1.5273423	1.1732982
C5	0.9869312	1.7802033	1.4261593
C6	0.73071474	1.5239868	1.1699429
C7	0.7461752	0.0470969	0.30694714
C8	0.7821395	1.5754116	1.2213676
C9	0.50417596	0.28909615	0.064947896

C. Clustering chromosomes using 3-dimensional k-means clustering

The 3-dimensional k-means clustering approach using the overlap factor, coverage factor and average of both is executed to divide the ten chromosomes into k clusters. The parameter k is set to 3.

k=3

Chromosome no 0 In Cluster No. 0
Chromosome no 1 In Cluster No. 1
Chromosome no 2 In Cluster No. 2
Chromosome no 3 In Cluster No. 0
Chromosome no 4 In Cluster No. 0
Chromosome no 5 In Cluster No. 0
Chromosome no 6 In Cluster No. 0
Chromosome no 7 In Cluster No. 1
Chromosome no 8 In Cluster No. 0
Chromosome no 9 In Cluster No. 2

D. Suitability of Chromosomes

The suitability factor used in the fitness function can reduce the occurrence of the two bad kinds of membership functions shown in Figure. 5, where the first one is too redundant, and the second one is too separate.

Table 5. Suitability of Chromosomes

Chromosome No.	Suitability
0	9.695238
1	10.404762
2	10.088096
3	9.014286
4	9.038666
5	8.8125
6	9.041667
7	10.3626375
8	8.995671
9	10.146187

E. Fitness of Chromosomes

The fitness of each set of membership functions is evaluated by the number of large 1-itemsets generated by executing part of the proposed fuzzy mining algorithm. Using the number of large 1-itemsets a trade-off between execution time and rule interestingness can be achieved. Usually, a larger number of 1-itemsets will result in a larger number of all itemsets with a higher probability, which will thus usually imply more interesting association rules. The evaluation by 1-itemsets is, however, faster than that by all itemsets or interesting association rules.[2]

Table 6. Fitness of Chromosomes

Chromosome no.	Fitness
0	2.8880157
1	2.5949657
2	2.577295
3	3.1061807
4	3.0978024
5	3.177305
6	3.096774
7	2.6055143
8	3.1126082
9	2.562539

F. Multilevel Association Rule and Confidence

Fuzzy association rules (Second Level) and the confidence of those rules are obtained by the proposed system. The following two rules along with their confidence factors are output to the user.

IF MILKM AND BREADM THEN COOKIESM 30.0 %
IF MILKH AND BREADH AND COKKIESH THEN BEVERAGESH 40.0 %

XI. Comparison of Existing System and Proposed system

For clustering the parameter k was taken as 3 and comparison was done between the k-means algorithm and the 3-dimensional k-means clustering and the results obtained are as follows.
By looking at the Table 7 we find that the clusters obtained by using 3-dimensional k-means clustering are much better than the clusters formed by k –means clustering approach.

Table 7. Comparison of clusters formed

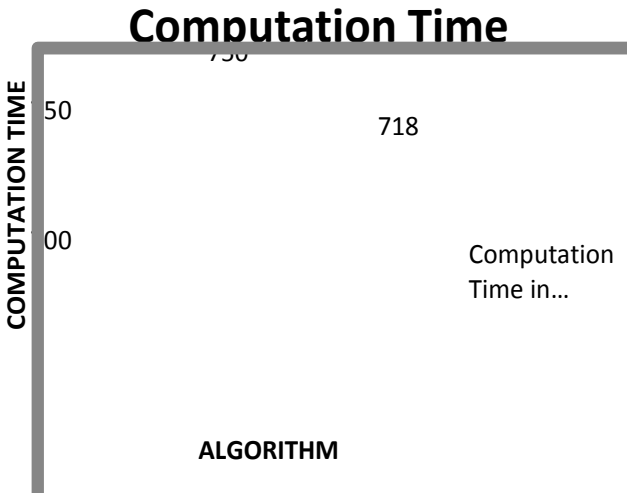
Clusters formed by 2-dimensional k-means clustering	Clusters formed by 3-dimensional k-means clustering
Chromosome no 0 In Cluster No. 1	Chromosome no 0 In Cluster No. 0
Chromosome no 1 In Cluster No. 1	Chromosome no 1 In Cluster No. 1
Chromosome no 2 In Cluster No. 1	Chromosome no 2 In Cluster No. 2
Chromosome no 3 In Cluster No. 1	Chromosome no 3 In Cluster No. 0
Chromosome no 4 In Cluster No. 1	Chromosome no 4 In Cluster No. 0
Chromosome no 5 In Cluster No. 1	Chromosome no 5 In Cluster No. 0
Chromosome no 6 In Cluster No. 1	Chromosome no 6 In Cluster No. 0
Chromosome no 7 In Cluster No. 1	Chromosome no 7 In Cluster No. 1
Chromosome no 8 In Cluster No. 1	Chromosome no 8 In Cluster No. 0
Chromosome no 9 In Cluster No. 1	Chromosome no 9 In Cluster No. 2

In order to check performance of the Association algorithms, we have applied the algorithm to item dataset. Comparison is done by considering the Computation Time.

Table 8. Comparison of Computation Time

Sr no.	Algorithm	Computation time
1	Existing system (using k-means clustering)	750 milliseconds
2	Proposed system (using 3- dimensional k-means clustering)	718 milliseconds

Following graph depicts comparison of Computational time of existing system and proposed system with respect to Computational time for database of 10 items and 20 transactions .X axis in Graph denotes algorithm and y axis denotes Computational time. Computational time is measured in terms of milliseconds.



Graph 1 . Comparison of Algorithm using K-means and 3-dimensional k-means Clustering

XII. Conclusion

This project presents an optimization method developed to deal with a data mining problem. Decision making in business sector is considered as one of the critical tasks. The objective is to provide a tool to help experts to find associations between the items bought by a customer in supermarket. An association rule may be more interesting if it reveals relationship among some useful concepts such as quantity of items bought by customer rather than which item is bought by the customer that is by using fuzzy mining. The favorite things of customers change all the time that is the membership functions should be adjusted dynamically, which can be done by genetic algorithm. To deal with all these aspects the fuzzy genetic association rule mining using modified k-means clustering is used. To achieve this, the algorithm has been developed in two parts. The first part finds out fit membership functions using genetic algorithm and modified k-means algorithm. The second part finds association rule (second level) using these fit membership functions and the confidence of that rule. This algorithm is tested on 10 items and 20 transactions.

Acknowledgment

The authors acknowledge immense help received from the scholars whose articles are cited and included in the references section of this manuscript. The authors are also grateful to the authors , editors and publishers of all those articles, journals and books from where the literature of this manuscript has been discussed. The authors also express their gratitude towards the anonymous referees for their very constructive, valuable suggestions and useful technical comments which led to a significant improvement of the paper. I would also like to thank Dr. J W Bakal for his

suggestions and mentorship.

References

[1] Chun-Hao Chen, Tzung-Pei Hong, "Cluster-Based Evaluation in Fuzzy-Genetic Data Mining",IEEE transactions on fuzzy systems, Vol. 16, No. 1, February 2008 249, pp. 249-262.

[2] T. P. Hong, C. H. Chen, Y. L. Wu, and Y. C. Lee, "A GA-based fuzzy mining approach to achieve a trade-off between number of rules and suitability of membership functions," *Soft Computing*, vol. 10, no. 11, pp. 1091–1101, 2006.

[3] Hung-Pin Chiu, Yi-Tsung Tang, "A Cluster-Based Mining Approach for Mining Fuzzy Association Rules in Two Databases", *Electronic Commerce Studies*, Vol. 4, No.1, Spring 2006, Page 57-74.

[4] Tzung-Pei Hong, Chan-Sheng Kuo, Sheng-Chai Chi, "Trade-Off Between computation time and number of rules for fuzzy mining from quantitative data", *International Journal of Uncertainty, Fuzziness and Knowledge-Based systems*, Vol. 9, No. 5, 2001, page 587- 604.

[5] M. Sulaiman Khan, Maybin Muyeba, Frans Coenen, "Fuzzy Weighted Association Rule Mining with Weighted Support and Confidence Framework", *The University of Liverpool, Department of Computer Science, Liverpool, UK.*

[6] H. Ishibuchi and T. Yamamoto, "Rule weight specification in fuzzy rule-based classification systems," *IEEE Trans. on Fuzzy Systems*, Vol. 13, No. 4, pp. 428-435, August 2005.

[7] Miguel Delgado, Nicolás Marín, Daniel Sánchez, and María-Amparo Vila," Fuzzy Association Rules: General Model and Applications", *IEEE transactions on fuzzy systems*, vol. 11, no. 2, April 2003

[8] Tzung-Pei Hong, Li-Huei Tseng and Been-Chian Chien, " Learning Fuzzy Rules from Incomplete Quantitative Data by Rough Sets".

[9] H. J. Zimmermann, "Fuzzy set theory and its applications", *Kluwer Academic Publisher*, Boston, 1991.

[10] Tzung-Pei Hong, Ming-Jer Chiang and Shyue-Liang Wang "Mining from Quantitative Data with Linguistic Minimum Supports and Confidences",2002 IEEE Proceedings.

[11] S. Yue, E. Tsang, D. Yeung, and D. Shi, "Mining fuzzy association rules with weighted items," in *Proc. IEEE Int. Conf. on Systems, Man and Cybernetics*, 2000, pp. 1906–1911.

[12] M. Kaya, R. Alhajj , *"Genetic algorithm based framework for mining fuzzy association rules",* 2004 Elsevier B.V.

[13] Jiawei Han, Micheline Kamber," Data mining: concepts and techniques", second Edition, Morgan Kaufmann publisher, 2006, pp.401-407

[14] R. Agrawal, T. Imielinksi and A. Swami, "Database mining: a performance perspective," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 5, No. 6, 1993, pp. 914-925.

[15] Tzung-Pei Hong, Ming-Jer Chiang, Shyue-Liang Wang, "Data Mining with Linguistic Thresholds", Int. J. Contemp. Math. Sciences, Vol. 7, 2012, no. 35, 1711 – 1725.

[16] Chun-Hao Chen, Tzung-Pei Hong, Vincent S.Tseng, *"A Brief Survey of Genetic- Fuzzy Data Mining Techniques",*International Conference on Advanced Information Technology , 2008.

Author Biographies

Shaikh Nikhat Fatma has received the M.E. degree in Computer Engineering from Pillai Institute of Information Technology, Engineering, Media Studies and Research, Mumbai University in 2012. She has received the B.E. degree in Computer Engineering from Lokmanya Tilak College of Engineering, Mumbai University in 2007. Her research interests include data mining, genetic algorithms and fuzzy theory.