A FUZZY FREQUENT PATTERN-GROWTH ALGORITHM FOR ASSOCIATION RULE MINING

A.H.M. Sajedul Hoque[1], Rashed Mustafa[1], Sujit Kumar Mondal[2] and Md. Al-Amin Bhuiyan[3]

[1]Dept. of Computer Science & Engineering,
University of Chittagong, Chittagong, Bangladesh
[2]Dept. of Computer Science & Engineering,
Islamic University, kushtia, Bangladesh
[3]Dept. of Computer Engineering,
King Faisal University, Al Ahsa, Saudi Arabia

ABSTRACT

Currently the number of tuples of a database of an enterprise is increasing significantly. Sometimes the associations among attributes in tuples are essential to make plan or decision for future for higher authority of an organization. The quantitative attributes in tuples must be split into two or more intervals. Due to the over and under-estimation problem closer to the boundary of classical logic, fuzzy logic has been used to make intervals for quantitative attribute. These fuzzy intervals are based on the generation of more realistic associations. This paper focuses on implication of association rules among the quantitative attributes and categorical attribute of a database employing fuzzy logic and Frequent Pattern (FP) - Growth algorithm. The effectiveness of the method has been justified over a sample database.

KEYWORDS

Data Mining, Association Rule Generation, Fuzzy Frequent-Pattern Growth Algorithm, Fuzzy Logic

1. INTRODUCTION

The amount of data kept in computer files and database is growing at a phenomenal rate. At the same time, the users of these data are expecting more sophisticated information from them [1]. Data mining is the nontrivial extraction of implicit, previously unknown and potentially useful information from data [2]. Data mining refers to the process that retrieves knowledge from large database. There are lots of data mining tasks such as association rule, regression, clustering and prediction [3]. Among these tasks association rule mining is most prominent. Association rules are used to retrieve relationships among a set of attributes in a database. There are lots of algorithms to generate association rules from a database, such as Apriori, Frequent Pattern Growth (FP-Growth), Eclat, Recursive Elimination etc. These produced association rules can be used to make decision or plan for future for an organization. This paper focuses on FP-Growth algorithm to generate fuzzy association rule from an employee database. The source database may contain Boolean or categorical or quantitative attributes. In order to generate association rule over quantitative attributes, the domain of quantitative attributes must be split into two or more

DOI : 10.5121/ijdkp.2015.5502

intervals. This paper explores the generation of association rules on quantitative attributes by employing fuzzy logic.

The rest of the paper is organized as follows. Section 2 describes the basics of fuzzy association rules. The FP-Growth algorithm is described in section 3. The process of mining fuzzy association rules from a sample database is presented in section 4. Section 5 illustrates the experimental results. And finally section 6 concludes the paper.

2. FUZZY ASSOCIATION RULES

Association rules represent a convenient and effective way to identify certain dependencies between attributes in a database [4]. Given a set of items $I = \{I_1, I_2, ..., I_m\}$ and a database of transactions $D = \{t_1, t_2, ..., t_n\}$ where $t = \{I_{i1}, I_{i2}, ..., I_{ik}\}$ and $I_{ij} \in I$, an association rule is an implication of the form $X \Rightarrow Y$ where $X, Y \subset I$ are sets of items called itemsets, and $X \cap Y = \phi$ [1]. An association rule is said to be Fuzzy Association Rule, if the both or one of itemsets in implication, i.e X and Y, are fuzzy or vague. Here, X is called antecedent, and Y consequent. For example $Old \Rightarrow High\ Salary$ represents that if an employee is Old, he also gets High Salary. Two important constraints are used to measure the interestingness of an association rule, such as support and confidence. The support of a rule, $X \Rightarrow Y$, is the ratio (in percent) of the records contain $X \cup Y$ to the total number of records in the database. The confidence of an associational rule, $X \Rightarrow Y$, is the ratio (in percent) of the number of records that contain $X \cup Y$ to the number of records that contain X [5]. The attributes of a database may be either Boolean or quantitative or categorical. The attribute whose domain is {0,1} or {yes,no} or {true, false} is called Boolean attribute. The attribute which has any numerical value is called quantitative attribute. Based on the types of values, the association rules can be classified as Boolean Association rules and quantitative association rules. The association rules with Boolean items is called Boolean association rule. For example, $Keyboard \Rightarrow Mouse$ [support = 20% and confidence = 60%], where both keyboard and mouse are Boolean items. The association rule with quantitative items is called quantitative association rules. For example $(Age = 26\ \ to\ \ 30) \Rightarrow (Car = 1,2)$ [support = 20% and confidence = 60%], where age and car are both quantitative attributes. If the number of values in the domain of quantitative attributes is very limited then association rules can be generated easily. But when the number of values increases, the domain must be split into intervals to generate association rules. Usually quantitative association rules are generated using classical logic, which is also called classical association rules. In classical quantitative association rule, the intervals are constructed using classical or crisp set theory. In classical set theory, very precise bounds separate the elements that belong to a certain set from the elements outside the set. Consequently in classical interval, any value of a quantitative attribute belongs to one interval with membership degree 1 and in other intervals the membership degree is 0. Another special feature of classical intervals is that there is no overlap among intervals. For example consider the domain of age attribute of a database is age={22, 27, 33, 43, 54, 67, 70, 80} and it is split into three intervals named low, medium and high, where the ranges are 0 to 39, 40 to 59 and 60 to 80 respectively. So low = {(22,1),(27,1),(33,1),(43,0),(67,0),(70,0),(80,0)}, medium = { (22,0), (27,0), (33,0), (43,1), (67,1), (70,0), (80,0)} and high = {(22,0),(27,0),(33,0),(43,0),(67, 0),(70, 1),(80, 1)} respectively, where each set consists of ordered pairs, the first element of the ordered pair is the element of universal set age and the second element is the membership degree of that element. Though the difference between value 2 and 39 is 37, both value is considered in same interval, low. But the values 39 and 40 belong to different intervals, low and high, in spite of the difference being 1. So creating intervals using classical logic leads over-estimation and under-estimation of values closer to the boundary which consequently affect on the generation of association rules. To alleviate these hard boundary problems, fuzzy set theory is used to split the quantitative attributes. In fuzzy set, the membership degree of each element is any value in between 0 and 1 including 0 and 1 [6]. If the ranges for low, medium and high are respectively 0 to 40, 30 to 60 and 55 to 80, according to the fuzzy set theory the intervals are defined as low = {(22,.8),(27,.6),(33,.2),(43,0),(67,0),(70,0),(80,0)}, medium = {(22,0),(27,0),(33,.4),(43,.8), (67,0),(70,0),(80,0)} and high = {(22, 0),(27, 0),(33, 0),(43, 0),(67, .6),(70, .8),(80, 1)}, where the membership degree of each fuzzy class is characterized by using one of membership functions, such as S-shaped, Z-shaped, Π-shaped, Triangular etc. These fuzzy intervals lead to the generation of more meaningful and right association rules than classical intervals.

3. MINING OF ASSOCIATION RULES USING FP-GROWTH ALGORITHM

Currently FP-Growth algorithm is one of the fastest algorithms to mine association rules. This algorithm generates frequent itemsets and it does not create huge amount of candidate itemsets like Apriori algorithm. Before applying FP-growth algorithm on source database, the source database must be pre-processed. During the pre-processing of the database, the frequency of all items are determined by scanning the whole database, then all infrequent items are discarded from each transaction and finally the items of each transaction are sorted according to their frequency. Then from the pre-processed database a FP-tree is constructed. FP-tree is a highly compact representation of the original database, which is assumed to fit into the main memory. Algorithm 1 shows the algorithm of the construction of FP-tree [7,8].

Algorithm 1 (FP-tree construction)
Input: A transaction database DB and a minimum support threshold ξ.
Output: Its frequent pattern tree, FP-Tree
Method: The FP-tree is constructed in the following steps.

1. Scan the transaction database DB once. Collect the set of frequent items F and their supports. Discard all infrequent items from each transaction of DB. Sort F in support descending order as L, the list of frequent items.
2. Create the root of an FP-tree, T, and label it as "null", for each transaction in DB.
3. Select and sort the frequent items in transaction according to the order of L.
4. Let the sorted frequent item list in transaction be [p|P], where p is the first element and P is the remaining list. Call insert_tree ([p|P], T).

Procedure insert_tree ([p|P], T)

1. {
2. If T has a child N such that N.item-name = p.item-name, then increment N's count by 1;
3. Else do

i. {
ii. create a new node N;
iii. N's count = 1;
iv. N's parent link be linked to T;
v. N's node-link be linked to the nodes with the same item-name via the node-link structure;
vi. }

4. If P is nonempty, Call insert_tree (P, N).
5. }

After having the FP-tree, all frequent itemsets are generated using Algorithm 2 [7,8].

Algorithm 2 (FP-growth)
Input: FP-tree constructed based on Algorithm 1, DB and a minimum support threshold ξ.
Output: The complete set of frequent patterns.
Method: Call **FP-growth (FP-tree, null)**

Procedure FP-growth (Tree, α)

1. {
2. if Tree contains a single path P **then for each** combination (denoted as β) of the nodes in the path P **do**
   generate pattern $\beta \cup \alpha$ with support= minimum support of nodes in β;
3. **else for each** a_i in the header of Tree (in reverse order) **do**
   i. {
   ii. generate pattern $\beta = a_i \cup \alpha$ with support = a_i.support;
   iii. construct β's conditional pattern base and then β's conditional FP-tree Tree β;
   iv. **if** Tree $\beta \neq \phi$
   v. **then** call **FP-growth (Tree β, β)**
   vi. }
4. }

This algorithm is accomplished by traversing from bottom node of FP-tree to root node. During traversing at each level of the tree the FP-Growth algorithm checks if the node has a single path. If it is not, at first conditional pattern bases of that node are determined, secondly all infrequent items are discarded from the determined pattern bases and finally a FP-tree is constructed. This process is repeated until a single path is found. If a single path is found, all combinations of the path including candidate node are determined. These combinations are the desired frequent itemsets of the desired node. After getting frequent all itemsets, association rules are generated whose confidence value is greater than threshold value. In the produced association rule, antecedent consists of any number of items of frequent itemsets and consequent consists of only one item. This paper explores the FP-growth algorithm on a sample employee database, which is illustrated in table 1 [9].

Table 1. Employee database

TID	Age	Salary (Tk.)	Education	Service Years
1	23	12000	High School	5
2	28	15800	Bachelor	3
3	28	17000	Master	1
4	30	21300	Master	2
5	30	9500	High School	1
6	37	28000	PHD	1
7	39	20000	Bachelor	8
8	41	36500	PHD	11
9	44	32000	Master	15
10	46	15000	High School	23

Here Age, Salary and Service Years are all quantitative attributes and Education is categorical attribute. It is mentioned that it is necessary to split the quantitative attributes into two or more intervals to generate association rules. In this paper, Age attribute is split into Young (Y), Middle Aged (MA) and Old (O), Salary attribute is split into Low Salary (LSa), High Salary (HSa), Medium Salary (MSa) and finally Service Years are split into High Experienced (HEx), Medium Experienced (MEx) and Low Experienced (LEx). And the domain of categorical attribute, Education, is {High School (HSc), Bachelor (Ba), Master (Ms), PHD}. In this paper, fuzzy association rules are generated over these quantitative attributes and categorical attributes.

4. Generation of Fuzzy Association Rule

Generation of association rules becomes easier, when the domain of quantitative attributes is limited. But for large domain, the quantitative attribute must be split into two or more intervals. In fuzzy association rules, the intervals are constructed employing fuzzy logic. In fuzzy intervals, one element belongs to more than one interval with different degrees. The process to generate fuzzy association rule of table 1 is described below step by step:

4.1 Constructing Fuzzy Intervals

Intervals of quantitative attributes can be constructed either by consulting with an expert or by applying statistical approach. In this paper, each quantitative attribute is split into three fuzzy intervals using statistical approach and, which are illustrated in figure 1[10].

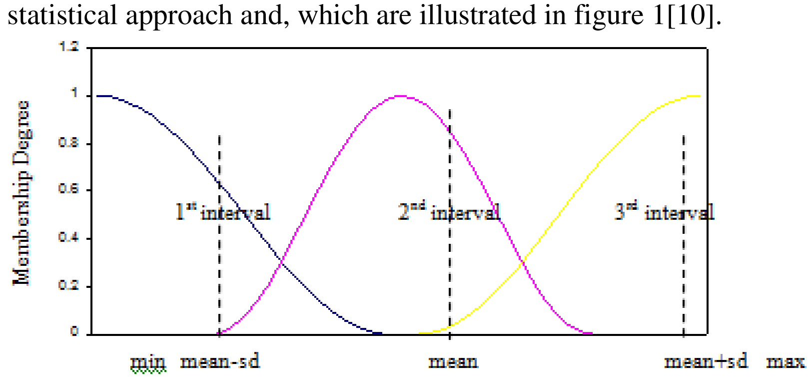

Figure 1. Values of Quantitative Attribute

1st interval: The lower border (lb) of 1st interval is the minimum value over the domain of the quantitative attribute. The higher border (hb) is computed using the mean and standard deviation (sd) of the values of quantitative attribute. The mathematical expression of lb and hb is shown in Eq. 1.

$$\left.\begin{aligned} lb &= \min(\mathit{quantitative\ \ attribute}) \\ hb &= \mathit{mean} - \frac{sd}{2} + \mathit{mean} \times \mathit{overlap} \end{aligned}\right\} \tag{1}$$

In the 1st interval Z-membership function is used to compute membership degree, which is shown in Eq. 2 [11].

$$f(x)_z = \frac{1}{2} + \frac{1}{2}\cos\left(\frac{x - lb}{hb - lb}\right)\Pi \tag{2}$$

2nd interval: The lower border (lb) and higher border (hb) of 2nd interval are computed using the Eq. 3.

$$\left.\begin{aligned} lb &= \mathit{mean} - \frac{sd}{2} - \mathit{mean} \times \mathit{overlap} \\ hb &= \mathit{mean} + \frac{sd}{2} + \mathit{mean} \times \mathit{overlap} \end{aligned}\right\} \tag{3}$$

This interval can be characterized using both S-membership and Z-membership function, which is expressed in Eq. 4.

$$\left.\begin{aligned} f(x)_S &= \frac{1}{2} + \frac{1}{2}\cos\left(\frac{\mathit{mean} - x}{\mathit{mean} - lb}\right)\Pi \quad lb \le x \le \mathit{mean} \\ f(x)_z &= \frac{1}{2} + \frac{1}{2}\cos\left(\frac{x - \mathit{mean}}{hb - \mathit{mean}}\right)\Pi \quad \mathit{mean} \le x \le hb \end{aligned}\right\} \tag{4}$$

3rd interval: Eq. 5 is used to compute the *lower border (lb)* and *higher border (hb)* of 3rd interval.

$$\left.\begin{aligned} lb &= \mathit{mean} + \frac{sd}{2} - \mathit{mean} \times \mathit{overlap} \\ hb &= \max(\mathit{quantitative\ \ attribute}) \end{aligned}\right\} \tag{5}$$

This interval has been characterized using the S-Membership function shown in Eq. 6.

$$f(x)_s = \frac{1}{2} + \frac{1}{2}\cos\left(\frac{hb - x}{hb - lb}\right)\Pi \tag{6}$$

Here the overlap which indicates the percentage of amount of overlapped values among intervals is 10%. After applying Eq. 1, Eq. 3 and Eq. 5, the fuzzy intervals of Age quantitative attribute will be [23, 34.16], [27.24, 41.96] and [35.04, 46] for Y, MA and O respectively, the fuzzy intervals of Salary attribute are [9500, 18362.64], [14220.64, 27199.36] and [23057.36, 36500] for LSa, MSa and HSa respectively and the fuzzy intervals for Service Years are [1, 4.01] for LEx, [2.61,11.39] for MEx and [9.99, 23] for HEx.

4.2. Preprocessing the Database

Before pre-processing the database, the quantitative attributes are mapped into fuzzy attributes using Eq. 2, Eq. 4 and Eq. 6 and categorical attributes should be mapped into Boolean attributes. If any value x belongs to any Boolean attribute X, i.e $x \in X$, x will be mapped into 1, otherwise it will be 0. The overall mapping table from the quantitative attribute of table 1 is shown in table 2. If the domains of a database containing three attributes are D1, D2 and D3, the database will be a subset of $D_1 \times D_2 \times D_3$ [12]. In classical database, the membership degree of each datum of each domain is either 0 or 1. But in fuzzy database the degree may be in between 0 and 1. The membership degree of each tuple is the minimum membership value over all membership degrees of that tuple. The database contains only the tuples containing non-zero membership degree. After applying this concept, the number of rows of classical database and fuzzy database of given employee database are 10 and 28 respectively. Table 3 is the obtained fuzzy database. The frequency of each item is computed by summing the membership degree of rows containing that item. For example the frequency of Y (Young) is (.57+.19+.02+.25+.02+.06+.58+.31+.31) or 2.31. The frequencies of all items in table 3 are shown in table 4. Here the total membership degree of all tuples in table 3 is 5.44. The support of each item is determined by summing all membership degrees of that item in every transaction. For example, the support of item Young (Y), i.e support (Y), is (.57+.19+.02+.25+.02+.06+.58+.31+.31) or 2.31. In this paper, the assumed minimum support and minimum confidence are respectively 20% and 70%. So the only infrequent items in table 4 are HSa and PHD. After discarding the infrequent items from every transaction and sorting the items in each transaction according to the descending order of their frequency in the database, the Pre-Processed database shown in table 5 is found.

Table 2. Mapping Table of Employee Database

TID	Age			Salary			Education				Service Years		
	Y [23, 34.16]	MA [27.24, 41.96]	O [35.04, 46]	LSa [9500, 18362.64]	MSa [14220.64, 27199.36]	Hsa [23057.36, 36500]	HSc	Ba	Ms	PHD	LEx [1, 4.01]	MEx [2.61, 11.4]	Hex [9.99, 23]
1	1	0	0	0.82	0	0	1	0	0	0	0	0.57	0
2	0.58	0.03	0	0.19	0.98	0	0	1	0	0	0.25	0.02	0
3	0.58	0.03	0	0.06	0.62	0	0	0	1	0	1	0	0
4	0.31	0.31	0	0	0.98	0	0	0	1	0	0.75	0	0
5	0.31	0.31	0	1	0	0	1	0	0	0	1	0	0
6	0	0.76	0.08	0	0	0.3	0	0	0	1	1	0	0
7	0	0.35	0.29	0	0.79	0	0	1	0	0	0	0.88	0
8	0	0.04	0.57	0	0	1	0	0	0	1	0	0.02	0.01
9	0	0	0.92	0	0	0.75	0	0	1	0	0	0	0.32
10	0	0	1	0.32	0.61	0	1	0	0	0	0	0	1

Table 3. Fuzzy Database

Sl. No	Tuples	Membership Degree
1	Y LSa HSc MEx	0.57
2	Y LSa Ba LEx	0.19
3	Y LSa Ba Mex	0.02
4	Y MSa Ba LEx	0.25
5	Y MSa Ba Mex	0.02
6	MA LSa Ba LEx	0.03
7	MA LSa Ba Mex	0.02
8	MA MSa Ba LEx	0.03
9	MA MSa Ba MEx	0.02
10	Y LSa Ms LEx	0.06
11	Y LSa Ms LEx	0.58
12	MA LSa Ms LEx	0.03
13	MA MSa Ms LEx	0.03
14	Y MSa Ms LEx	0.31
15	MA MSa Ms LEx	0.31
16	Y LSa HSc LEx	0.31
17	MA LSa HSc LEx	0.31
18	MA HSa PHD LEx	0.3
19	O HSa PHD LEx	0.08
20	MA MSa Ba MEx	0.35
21	O MSa Ba MEx	0.29
22	MA HSa PHD MEx	0.02
23	O HSa PHD MEx	0.02
24	MA HSa PHD HEx	0.02
25	O HSa PHD HEx	0.02
26	O HSa Ms HEx	0.32
27	O LSa HSc HEx	0.32
28	O MSa HSc HEx	0.61

Table 4. Frequency of Items

Sl. No	Items	Support
1	Y (Young)	2.31
2	MA (Middle Aged)	1.47
3	O (Old)	1.66
4	LSa (Low Salary)	2.44
5	MSa (Medium Salary)	2.22
6	HSa (High Salary)	0.78
7	HSc (High School)	2.12
8	Ba (Bachelor)	1.22
9	Ms (Master)	1.64
10	PHD	0.46
11	LEx (Low Experienced)	2.82
12	MEx (Medium Experienced)	1.33
13	HEx (High Experienced)	1.29

Table 5. Pre-Processed Database

Sl. No	Items	(ordered) Frequent Items
1	Y LSa HSc MEx	LSa Y HSc Mex
2	Y LSa Ba LEx	LEx LSa Y Ba
3	Y LSa Ba MEx	LSa Y MEx Ba
4	Y MSa Ba LEx	LExY MSa Ba
5	Y MSa Ba MEx	Y MSa MEx Ba
6	MA LSa Ba LEx	LEx LSa MA Ba
7	MA LSa Ba MEx	LSa MA MEx Ba
8	MA MSa Ba LEx	LEx MSa MA Ba
9	MA MSa Ba MEx	MSa MA MEx Ba
10	Y LSa Ms LEx	LEx Y LSa Ms
11	Y LSa Ms LEx	LEx Y LSa Ms
12	MA LSa Ms LEx	LEx LSa Ms MA
13	MA MSa Ms LEx	LEx MSa Ms MA
14	Y MSa Ms LEx	LEx Y MSa Ms
15	MA MSa Ms LEx	LEx MSa Ms MA
16	Y LSa HSc LEx	LEx LSa Y HSc
17	MA LSa HSc LEx	LEx LSa HSc MA
18	MA HSa PHD LEx	LEx MA
19	O HSa PHD LEx	LEx O
0	MA MSa Ba MEx	MSa MA MEx Ba
21	O MSa Ba MEx	MSa O MEx Ba
22	MA HSa PHD MEx	MA Mex
23	O HSa PHD MEx	O Mex
24	MA HSa PHD Hex	MA Hex
25	O HSa PHD Hex	O Hex
26	O HSa Ms Hex	O Ms Hex
27	O LSa HSc Hex	LSa HSc O Hex
28	O MSa HSc Hex	MSa HSc O Hex

4.3. Building the Fuzzy FP-tree

An FP-tree is basically a prefix tree in which each path represents a set of transactions that share the same prefix. Then the procedure insert_tree ([p|P], T) is applied on ordered frequent items in table V to build up FP-tree, which is shown in fig. 2.

According to the procedure, after crating a root termed “null”, the first branch is constructed for transaction <LSa Y HSc MEx :.57>, where four new nodes are created for items Y, LSa, HSc and MEx. And the node LSa is linked as the child of null, Y is linked as the child of LSa, HSc is linked as the child of node Y, finally the node MEx is linked as the child of node HSc. As the next transaction <LEx LSa Y Ba : .19> does not share common prefix with previous transaction, another branch has been created for the second transaction. Since the 1st transaction and 3rd transaction has common prefix <LSa Y>, it is not necessary to create new nodes for LSa and Y items. In that case, the membership degree of LSa and Y has been added with the previous degree. At node Y, another new branch <MEx Ba> is constructed, where MEx is child of Y and Ba is the child of MEx. In this way all transactions are embedded into FP-tree. During the construction of FP-tree, a header file is build, in which each item points to its first occurrence in the tree via a node-link. Nodes with the same item name are linked in sequence via such node-links.

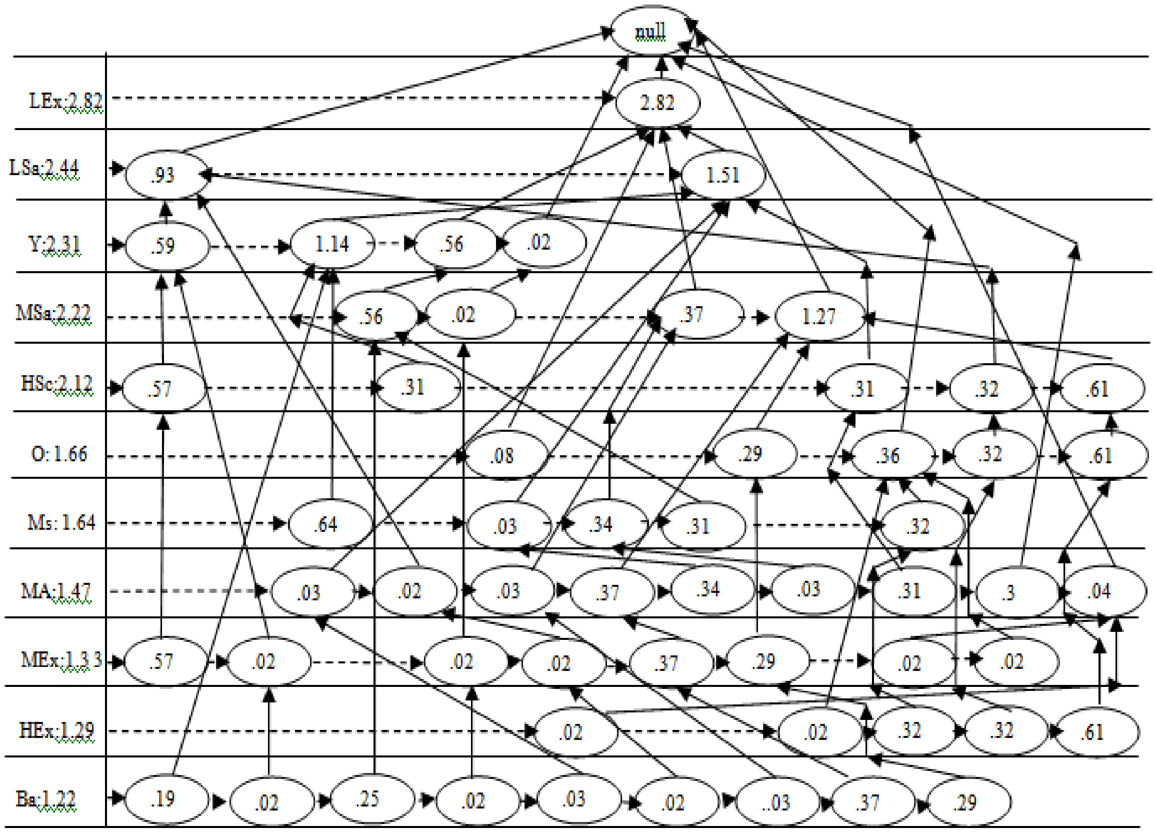

Figure 2. Fuzzy FP-tree

4.4. Generation Fuzzy Frequent Itemsets

To produce frequent itemsets, the Algorithm 2 is applied on the produced FP-tree. The process of this algorithm starts from the child node, Ba, and finishes at the root node “null”. Here the node Ba has nine paths and thus nine prefix paths: <LEx LSa Y:.19>, <LSa Y MEx:.02>, <LEx Y MSa:.25>, <Y MSa MEx:.02>, <LEx LSa MA:.03>, <LSa MA MEx:.02>, <LEx MSa MA:.03>, <MSa MA MEx:.37> <MSa O MEx:.29>, which are also called the conditional pattern bases of Ba. Since all items in conditional pattern bases of Ba are infrequent, no FP-tree can be constructed. Thus there is no frequent itemsets including item Ba. Similarly the conditional pattern bases of the item Hex are <MA: .02>, <O:.02>, <O Ms:.32>, <O HSc LSa:.32> and <O HSc MSa:.61>. Since the only frequent item is O with support 1.27, the frequent itemsets including O item is <O HEx: 1.27>. In this way all frequent itemsets are determined from FP-tree shown in 2, which are listed in table 6.

Table 6. Frequent Itemsets

Item	Conditional Pattern Bases	Frequent itemsets
Ba	{<LEx LSa Y:.19>, <LSa Y MEx:.02>, <LEx Y MSa:.25>, <Y MSa MEx:.02>, <LEx LSa MA:.03>, <LSa MA MEx:.02> <LEx MSa MA:.03> <MSa MA MEx:.37> <MSa O MEx:.29>}	ϕ
HEx	{<MA:.02>,<O:.02>, <O Ms:.32>, <O HSc LSa:.32> <O HSc MSa:.61>}	<O HEx:1.27>
MEx	{<Y LSa HSc:.57>, <LSa Y:.02>, <Y MSa:.02>, <LSa MA:.02>, < MSa MA:.37>, <MSa O:.29>, <MA:.02>,<O:.02>}	ϕ
MA	{<LEx LSa:.03>, <LSa:.02>, <LEx MSa:.03>, <MSa:.37>, < LEx LSa Ms:.34>, < LEx MSa Ms:.03>, < LEx LSa HSc:.31>}	ϕ
Ms	{<LEx LSa Y:.64>, <LEx LSa:.03>, <LEx MSa:.34>, <LEx Y MSa :.31>, <O:.32>}	< LEx Ms:1.32>
O	{<LEx:.08>,<MSa:.29>, <LSa HSc:.32>, < MSa HSc:.61>}	ϕ
HSc	{<LSa Y:.57>, <LEx LSa Y:.31>, < LEx LSa:.31>, <LSa:.32>, <MSa:.61>}	< LSa HSc:1.51>
MSa	{<LEx Y:.56>,<Y:.02>, <LEx:37>}	ϕ
Y	{<LSa:.59>, < LEx LSa:1.14>, <LEx:.56>}	<LEx LSa Y:1.14>
LSa	{<LEx:1.51>}	< LEx LSa:1.51>
LEx	ϕ	ϕ

4.5. Generation Fuzzy Association Rules

From the produced frequent itemsets, the interesting association rules are generated. The confidence of association rule, $X \Rightarrow Y$, is determined using the Eq. 7.

$$confidence(X \Rightarrow Y) = \frac{\sup port(X \cup Y)}{\sup port(X)} \times 100\% \quad (7)$$

During the generation of association from frequent itemsets, any one item of the frequent itemsets is placed as consequent and the rest of the items are placed as antecedent in association rule. Then the confidence value is determined using equation 4. If the determined confidence value of the rule is greater than the minimum confidence, then the association rule is interesting, otherwise not. For example, consider the frequent itemset <O HEx:1.25>, where the support of this association is 1.25. So the candidate association rules are $O \Rightarrow HEx$ and $HEx \Rightarrow O$. The confidences are: $confidence(O \Rightarrow HEx) = \frac{\sup port(O \cup HEx)}{\sup port(O)} \times 100\% = 76.5\%$, $confidence(HEx \Rightarrow O) = \frac{\sup port(HEx \cup O)}{\sup port(HEx)} \times 100\% = 98.45\%$ Since the minimum confidence value is 70%, the interesting association rules are $O \Rightarrow HEx$ and $HEx \Rightarrow O$ and the confidence values are 76.5% and 98.45%. The first rule indicates that if an employee is old, he or she posses high experienced and the possibility is 76.5%. The second association rule represents that if an employee has high experienced, he or she will be old and again the confidence is 98.45%.

5. EXPERIMENTAL RESULTS

The generated association rules of employee database in table I employing fuzzy logic and FP-growth algorithm are listed in table 7, which represents overall behaviour of the organization over employees.

Table 7. Experimental Results

Sl No	Generated Fuzzy Association Rules	Confidence
1	$O \Rightarrow HEx$	76.5%
2	$HEx \Rightarrow O$	98.45%
3	$LEx \Rightarrow Ms$,	46.8%
4	$Ms \Rightarrow LEx$	80.5%
5	$LSa \Rightarrow HSc$	61.89%
6	$HSc \Rightarrow LSa$,	71.23%
7	$LEx\ LSa \Rightarrow Y$	75.5%
8	$LEx\ Y \Rightarrow LSa$	67.06%
9	$LSa\ Y \Rightarrow LEx$	65.9%
10	$LEx \Rightarrow LSa$	53.55%
11	$LEx \Rightarrow LSa$	61.89%

6. CONCLUSION

This paper presents implication of fuzzy association rules over three quantitative attributes and a categorical attribute, where quantitative attributes have been split into three intervals using fuzzy logic. These association rules represent the behaviour of the employee of an organization. Through these association rules one can easily know the salary structure of employee of that organization with age, job experiences and education qualification. Theses association rules can be used to classify the employee to give some opportunities, such as salary increment, promotion etc. Since fuzzy intervals are out of over-estimation and under-estimation of value closer to the boundary, it leads to produce more realistic associations than classical logic. In this paper, a small database has been used. The experimental result must be more significant if a real database is used.

ACKNOWLEDGMENT

The authors would like to express their heartiest felicitation and sense of gratitude to Professor Dr. Md. Nurul Islam, Registrar, Northern University Bangladesh for his encouragement regarding their research on Data Mining. They are also grateful to Professor Dr. Mir. Md. Akramuzzaman for his nice cooperation and constant support.

REFERENCES

[1] M.H. Dunham,"Data Mining", Pearson Education, Delhi, 6th impression, 2009.

[2] Frawley, William J.; Piatetsky-Shapiro, Gregory and Matheus, Christopher J., "Knowledge Discovery in Databases: An Overview " AAAI/MIT Press, 1992, pp-1-27 .

[3] Fayyad; Piatetsky-Shapiro and Smyth, "From Data Mining to Knowledge Discovery in Database", AI Magazine, 1996.

[4] M. De Cock, C. Cornelis, E.E. Kerre: "Elicitation of fuzzy association rules from positive and negative examples". Fuzziness and Uncertainty Modelling Research Unit, Department of Applied Mathematics and Computer Science,Ghent University, Belgium, 19 August ,2004.

[5] Rakesh Agrawal; Tomasz Imielinski and Arun N. Swami, "Mining Association Rules Between Sets of Items in Large Databases", Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, pp. 207-216, Washington, D.C., May 1993.

[6] George J. Klir and Bo Yuan: Fuzzy Sets and Fuzzy Application: Theory and Applications,Prentice Hall of India Private Limited, 2005

[7] Jiawei Han and Micheline Kamber, "Data Mining: Concepts and Techniques", China MachinePress, 2001.8, pp. 158-161.

[8] Jiawei Han, Jian Pei and Yiwen Yin., "Mining Frequent Patterns without Candidate Generation", ACM-SIGMOD Int. Conf. Management of Data (SIGMOD'00), pp 1-12, Dallas,TX, May 2000.

[9] A. H. M. Sajedul Hoque, Sujit Kumar Mondol, Tassnim Manami Zaman, Dr. Paresh Chandra Barman, Dr. Md. Al-Amin Bhuiyan, "Implication of Association Rules Employing FP-Growth Algorithm for Knowledge Discovery", 14th International Conference on Computer and Information Technology 2011 (ICCIT 2011), 22-24 December, Dhaka, Bangladesh.

[10] Bakk. Lukas Helm, "Fuzzy Association Rules An Implementation in R", Master Thesis, Vienna University of Economics and Business Administration, 2007.

[11] Tassnim Manami Zaman, A. H. M. Sajedul Hoque, Md. Al-Amin Bhuiyan, "Knowledge Discovery and Intelligent Query Employing Fuzzy Logic", The International Conference on Mathematics and Computer Science 2011 (ICMCS 2011), Chennai, India.

[12] Abraham Silberchatz, Henry F. Korth, S. Sudashan, "Database System Concepts", Mc Graw Hill, 5th Edition.