World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:4, No:5, 2010

Applying Fuzzy FP-Growth to Mine Fuzzy Association Rules

Chien-Hua Wang, Wei-Hsuan Lee, Chin-Tzong Pang,

Abstract—In data mining, the association rules are used to find for the associations between the different items of the transactions database. As the data collected and stored, rules of value can be found through association rules, which can be applied to help managers execute marketing strategies and establish sound market frameworks. This paper aims to use Fuzzy Frequent Pattern growth (FFP-growth) to derive from fuzzy association rules. At first, we apply fuzzy partition methods and decide a membership function of quantitative value for each transaction item. Next, we implement FFP-growth to deal with the process of data mining. In addition, in order to understand the impact of Apriori algorithm and FFP-growth algorithm on the execution time and the number of generated association rules, the experiment will be performed by using different sizes of databases and thresholds. Lastly, the experiment results show FFP-growth algorithm is more efficient than other existing methods.

Keywords—Data mining, association rule, fuzzy frequent pattern growth.

I. INTRODUCTION

DATA mining is a methodology for the extraction of new knowledge from data. This knowledge may relate to problem that we want to solve [11]. Thus, data mining can ease the knowledge acquisition bottleneck in building prototype systems [6]. If data mining extracting can effectively be applied on all varieties of analysis, it will assist the process of decision-making in business.

In common transactions, association rules ($X \to Y$) is the most popular mean to be applied. The purpose is to search for the relation that exists among items of database. The relation reflects that when items (X) appear, other items (Y) are likely to appear as well [4]. For instance, when a customer purchases bread, one might also get milk along with it. Thus association rules can assist decision makers to scope out the possible items that are likely to be purchased by consumers. Meanwhile, it facilitates planning marketing strategies [2].

This paper mainly focuses on the association rule technique. In the conventional association rule algorithm, scanning database takes enormous time particularly when one uses Apriori algorithm, which often affects the efficiency in data mining. To solve the drawback aforementioned, Han et al. [5] proposed a mining method, called Frequent-Pattern growth (FP-growth), which does not need to generate candidate itemsets and is considered more efficient. FP-growth is constructed by reading the data set one transaction at a time and mapping each transaction onto a path in a Frequent Pattern-tree (FP-tree). Since different transactions can have several items in common at the same time, their paths may overlap. The more paths overlap with one another, the more compression we can achieve by using the FP-tree structure. If the size of the FP-tree is small enough to fit into the main memory, this will allow us to extract frequent itemsets directly from the structure in memory instead of making repeated passes over the data stored on disk [13]. Therefore, it does not need to generate candidate itemset, all it needs is to scan the database twice.

Additionally, in regard to the matter of decision making, one has to take user's perception and cognitive uncertainty of subjective decisions into consideration. Zadeh proposed the fuzzy set theory [15] in 1965 to deal with cognitive uncertainty of vagueness and ambiguity. Since linguistic variables and linguistic values [16-18] can be described with fuzzy concepts to subjectively correspond with the possible cognition of a decision maker, they are handy in carrying out analysis of decisions-making. Thus, fuzzy data mining has recently become an important research matter.

Therefore, this paper proposes a fuzzy data mining method - Fuzzy Frequent Pattern-growth (FFP-growth) algorithm, which treats each item from a transaction database as a linguistic variable, and each linguistic variable is partitioned based on its linguistic value. In so doing, the natural language can be utilized to fully explain fuzzy association rules. There are two phases in the proposed method. One is to find large 1-itemsets by scanning the database once, and the other is to establish a fuzzy FP-tree by scanning the database twice. Then, one conditional pattern base and one conditional fuzzy FP-tree will be extracted from each node in a fuzzy FP-tree to generate the fuzzy association rules.

The remaining parts of this paper are organized as follows. Some literature reviews including Apriori algorithm, FP-growth algorithm and Fuzzy partition method will be in Section 2. The proposed algorithm for mining fuzzy association rules will be described in Section 3. An example to illustrate the proposed algorithm is given in Section 4. The experiment results will be explained in different sizes of databases, which two fuzzy association rules and minimum fuzzy support (FS) will be demonstrated to understand the impact will be in Section 5. The conclusion is then given in Section 6.

Chien-Hua Wang is with the Department of Information Management, Yuan Ze University, Taoyuan, 32003, Taiwan, ROC (e-mail: thuck@saturn.yzu.edu.tw).
Wei-Hsuan Lee is with the Department of Information Management, Yuan Ze University, Taoyuan, 32003, Taiwan, ROC (e-mail: whlee@saturn.yzu.edu.tw).
Chin-Tzong Pang is with the Department of Information Management, Yuan Ze University, Taoyuan, 32003, Taiwan, ROC (e-mail: imctpang@saturn.yzu.edu.tw).

II. Literature Reviews

A. Apriori algorithm

The most representative method of associations was proposed as Apriori algorithm [1] by Agrawal et al. in 1994. In Apriori algorithm process, two steps are included. Fist, it finds out the satisfied frequent itemsets of minimum support; second, it finds out the satisfied rules of minimum confidence. In other words, we are used to find out information of frequent itemset and mine all association rules. Apriori algorithm is continuously repeated to scan database, find out all frequent itemsets, until it does not produce new candidate itemsets. Apriroi algorithm does not filter prior candidate itemsets, so that reduces the amount of candidate itemsets to scan. Therefore, it needs many times to complete scanning a database. In implementing efficiency, Apriori algorithm is not completely efficient.

B. FP-growth algorithm

The new concept of FP-growth algorithm was proposed by Han et al.[5], it can be one of the representations of the itemsets which do not require candidate generations. It does not need association length to proceed phases which generate candidate itemsets in Apriori algorithm. However, mining with Apriori algorithm does not archive the goal efficiently because it may need many times to scan database and generate a lot of candidate itemsets. Therefore, FP-growth proceeds the first scan in transaction database, it then later filters the frequent itemsets and gradually increases support. Next, in the second scan, establish a FP-tree structure by the transaction database. Then, use a Header table to allocate each item node in FP-tree, each item of tree will link each other. Last, a Header table mines conditional pattern tree which finds out all frequent itemsets in recursive method. It is a very efficient and memory saving algorithm[13].

C. Fuzzy partition method

The concepts of linguistic variables were proposed by Zadeh [16-18] and it is reasonable that we view each attribute as a linguistic variable. Formally, a linguistic variable is characterized by a quintuple [12,19] denoted by $(x, T(x), U, G, M)$. Here, in which x is the name of the variable; $T(x)$ denotes the set of names of linguistic values or terms, are linguistic words or sentences in a natural language [3]; U denotes a universe of discourse; G is a syntactic rule for generating values of x; and M is a semantic rule for associating a linguistic value with a meaning.

Using the simple fuzzy partition method, each attribute can be partitioned by various linguistic values. The simple fuzzy partition methods have been widely used in pattern recognition and fuzzy reason. For example there are the applications to pattern classification by Ishibuchi et al. [9], to fuzzy neural networks by [10], and so the fuzzy rule generation by [14].

In the simple fuzzy partition methods, K various linguistic values are defined in each quantitative attribute. K is also pre-specified before executing the proposed method. Triangular and trapezoid membership functions are usually used for the

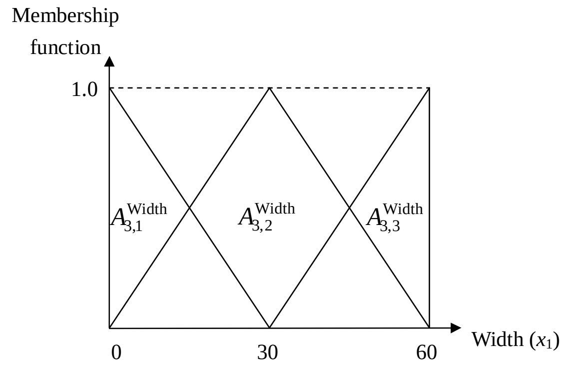

Fig. 1: $K = 3$ for "Width"

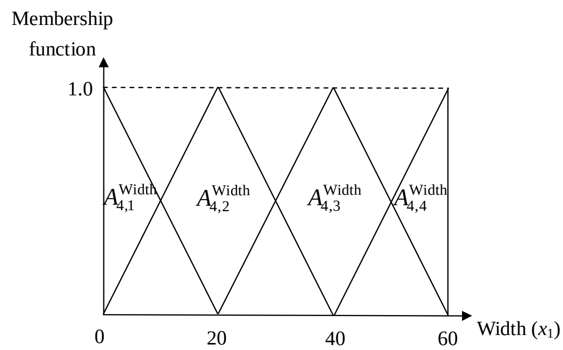

Fig. 2: $K = 4$ for "Width"

linguistic values. For example, $K = 3$ and $K = 4$ for the attribute "Width" (denoted by x_1) that ranges from 0 to 60 are shown as Figs. 1 and 2, respectively. That is three (i.e., small, middle and large) and four (i.e., small, middle small, medium large and large) various linguistic values are defined in Figs. 1 and 2, respectively [7-8].

In the method, each linguistic value is actually viewed as a candidate 1-dim fuzzy grid. Then A^{Width}_{K,h_1} can be represented as follows [7-8]:

$$\mu^{Width}_{K,h_1}(x) = \max\{1 - \frac{|x - a^K_{h_1}|}{b^K}, 0\} \tag{1}$$

Where

$$a^K_{h_1} = \frac{m + (M - m)(i_k - 1)}{K - 1}, \tag{2}$$

$$b^K = \frac{(M - m)}{K - 1} \tag{3}$$

where i_k is a linguistic value of K linguistic values in the linguistic variable for $1 \leq i_k \leq K$, and M and m are the maximum value and the minimum value of the domain interval of width, respectively.

III. The proposed method

Due to the growth in information technology has been rapidly developed, it makes store and management in database be more important. In recent years, large database and data warehouse are applied vastly. In general transactions, huge data need to be analyzed, including useful information and data. By each transaction record, we can search for the correlation among one and another. This section explains that how

FFP-growth algorithm can be applied onto great transaction database and mine useful information. Furthermore, it allows decision makers or businesses implement great marketing strategies or market planning.

A. FFP-growth algorithm

Input:

a. A body of n quantitative transaction data;
b. Minimum fuzzy support (min FS);
c. Minimum fuzzy confidence (min FC);
d. A linguistic value on every linguistic value, θ

Output: A set of fuzzy association rules.

Step 1: First, we apply fuzzy partition method to partition items into attributes.

Step 2: Count the scalar cardinalities of every fuzzy region after partitioning as the counting value.

Step 3: Choose the maximum region from counting value of each item as feature value of item.

Step 4: During scan one, find out the correspondences of min FS large 1-itemset, and establish Header table.

Step 5: According to the Header table, we renew and build fuzzy set of transaction table.

Step 6: During scan two, we establish a Fuzzy FP-tree.

Step 7: The item of Header table is mined sequence in descending order, and established each node of FFP-tree of conditional pattern base. Establish a conditional FFP-tree for each conditional pattern base. Next, repeat FFP-tree process mining, and mine frequent pattern of conditional FFP-tree in ascending order. Finally, if the conditional FFP-tree is contained one path, it will list all patterns.

Step 8: Using the following substeps mining fuzzy association rules.

(a) List all frequent itemsets.
(b) Count the confidences for all frequent itemsets. If they satisfy the confidences, then fuzzy association rules are generated.

B. Research framework

The paper utilizes fuzzy partition method which each transaction item was treated as a linguistic variable that is a proper number to partition. Next, apply FP-Growth scan one to find out large 1-itemset. In scan two, a FFP-tree is established. Further, conditional pattern base and conditional FFP-tree are established, respective. Finally, it uses recursive method to find out all fuzzy association rules, shown in Fig. 3.

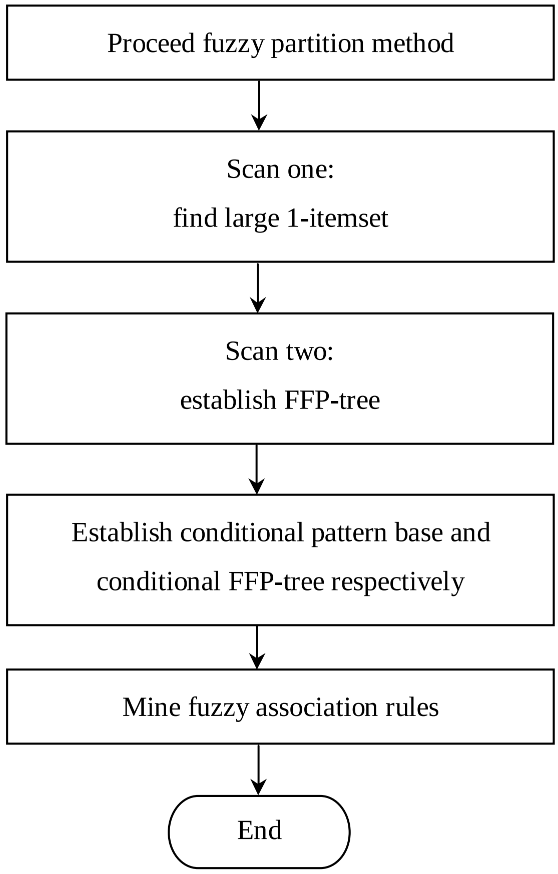

Fig. 3: The framework of this research

TABLE I: The ten transaction used in the example

TID	Items
1	(A, 1), (C, 3), (D, 6), (F, 9)
2	(B, 3), (C, 5), (D, 7), (E, 5), (F, 10)
3	(A, 8), (B, 4), (E, 3), (F, 13)
4	(A, 2), (B, 1), (C, 3), (D, 8), (E, 6), (F, 8)
5	(B, 5), (E, 10), (F, 7)
6	(A, 2), (C, 5), (D, 6), (E, 13)
7	(B, 3), (C, 6), (D, 7), (E, 12), (F, 10)
8	(A, 4), (B, 7), (C, 5), (D, 11), (E, 7), (F, 5)
9	(A, 3), (C, 11), (D, 10), (E, 9), (F, 4)
10	(A, 9), (B,1), (C, 11), (E, 8), (F, 13)

IV. An Example

In this section, an example is given to illustrate the proposed FFP-growth algorithm. This is an example to show how the proposed algorithm can be used to fuzzy association rules from a set of quantitative transactions. The data set includes ten quantitative transactions, as shown in Table 1. Each transaction is composed of a transaction identifier and items purchased. There are six items, respectively being A, B, C, D, E and F to be purchased. Each item is represented by a tuple (item name, item amount). In additional, the membership functions for item quantities are used three linguistic terms (Low, Middle and High) for all the items and are shown in Fig 4.

Here, assume that the predefined min FS and min FC are 0.28 and 0.60 respectively. By adopting membership functions of Fig 2, we only proceed with explanation and produce the result.

Step 1: Apply formula (1), (2) and (3) of fuzzy partition method which item attributes proceeded with partition, and get the result of fuzzy set after partition, shown as Table 2.

Step 2: The scalar cardinality of each fuzzy region in the transactions is calculated as the counting value. Take the fuzzy region A.Low as an example. Its scalar cardinality is (1.0 + 0

TABLE II: The fuzzy sets after partition

TID	Fuzzy Sets
1	$\left(\frac{1.000}{A.Low}\right), \left(\frac{0.667}{C.Low}+\frac{0.333}{C.Middle}\right), \left(\frac{0.167}{D.Low}+\frac{0.833}{D.Middle}\right), \left(\frac{0.667}{F.Middle}+\frac{0.333}{F.High}\right)$
2	$\left(\frac{0.667}{B.Low}+\frac{0.333}{B.Middle}\right), \left(\frac{0.333}{C.Low}+\frac{0.667}{C.Middle}\right), \left(\frac{1.000}{D.Middle}\right), \left(\frac{0.333}{E.Low}+\frac{0.667}{E.Middle}\right), \left(\frac{0.500}{F.Middle}+\frac{0.500}{F.High}\right)$
3	$\left(\frac{0.883}{A.Middle}+\frac{0.167}{A.High}\right), \left(\frac{0.500}{B.Low}+\frac{0.500}{B.Middle}\right), \left(\frac{0.667}{E.Low}+\frac{0.333}{E.Middle}\right), \left(\frac{1.000}{F.High}\right)$
4	$\left(\frac{0.833}{A.Low}+\frac{0.167}{A.Middle}\right), \left(\frac{1.000}{B.Low}\right), \left(\frac{0.667}{C.Low}+\frac{0.333}{C.Middle}\right), \left(\frac{0.167}{D.Middle}+\frac{0.833}{D.High}\right), \left(\frac{0.167}{E.Low}+\frac{0.833}{E.Middle}\right), \left(\frac{0.833}{F.Middle}+\frac{0.167}{F.High}\right)$
5	$\left(\frac{0.333}{B.Low}+\frac{0.667}{B.Middle}\right), \left(\frac{0.500}{E.Middle}+\frac{0.500}{E.High}\right), \left(\frac{1.000}{F.High}\right)$
6	$\left(\frac{0.833}{A.Low}+\frac{0.167}{A.Middle}\right), \left(\frac{0.333}{C.Low}+\frac{0.667}{C.Middle}\right), \left(\frac{0.167}{D.Low}+\frac{0.833}{D.Middle}\right), \left(\frac{1.00}{E.High}\right)$
7	$\left(\frac{0.667}{B.Low}+\frac{0.333}{B.Middle}\right), \left(\frac{0.167}{C.Low}+\frac{0.833}{C.Middle}\right), \left(\frac{1.000}{D.High}\right), \left(\frac{0.167}{E.Low}+\frac{0.833}{E.Middle}\right), \left(\frac{0.500}{F.Middle}+\frac{0.500}{F.High}\right)$
8	$\left(\frac{0.500}{A.Low}+\frac{0.500}{A.Middle}\right), \left(\frac{1.000}{B.Low}\right), \left(\frac{0.333}{C.Low}+\frac{0.667}{C.Middle}\right), \left(\frac{0.333}{D.Middle}+\frac{0.667}{D.High}\right), \left(\frac{1.000}{E.High}\right), \left(\frac{0.333}{F.Low}+\frac{0.667}{F.Middle}\right)$
9	$\left(\frac{0.667}{A.Low}+\frac{0.333}{A.Middle}\right), \left(\frac{0.333}{C.Middle}+\frac{0.667}{C.High}\right), \left(\frac{0.500}{D.Middle}+\frac{0.500}{D.High}\right), \left(\frac{0.667}{E.Middle}+\frac{0.333}{E.High}\right), \left(\frac{0.500}{F.Middle}+\frac{0.500}{F.High}\right)$
10	$\left(\frac{0.667}{A.Middle}+\frac{0.333}{A.High}\right), \left(\frac{1.000}{B.Low}\right), \left(\frac{0.333}{C.Middle}+\frac{0.667}{C.High}\right), \left(\frac{0.667}{E.Middle}+\frac{0.333}{E.High}\right), \left(\frac{1.000}{F.High}\right)$

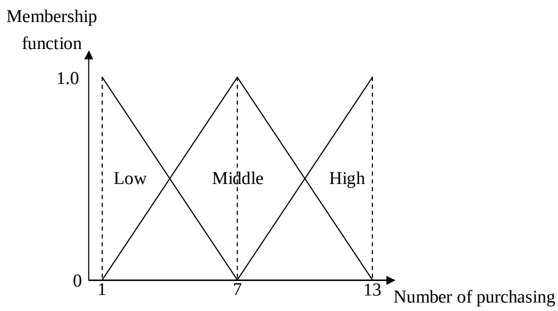

Fig. 4: The membership functions are used in the example

TABLE III: The countings of the fuzzy regions

Item	Count	Item	Count	Item	Count
A.Low	3.833	C.Low	2.50	E.Low	1.167
A.Middle	2.667	C.Middle	4.167	E.Middle	5.0
A.High	0.500	C.High	1.333	E.High	2.833
B.Low	4.167	D.Low	0.333	F.Low	0.833
B.Middle	2.833	D.Middle	5.333	F.Middle	4.667
B.High	0	D.High	1.333	F.High	3.50

+ 0 + 0.833 + 0 + 0.833 + 0 + 0.50 + 0.667 + 0) = 3.833. The step is repeated in the other regions and the results are shown in Table 3.

Step 3: The fuzzy region with the highest counting among the three possible regions for each item is found. Take item A as an example. Its counting is 3.833 for Low, 2.667 for Middle, and 0.500 for High. Since the counting for Low is the highest among the three countings, so the region A.Low is used to represent the item A. The step is repeated for the other items. Thus, ”Low” is chosen for B, ”Middle” is chosen for C, ”Middle” is chosen for D, ”Middle” is chosen for E and ”Middle” is chosen for F.

Step 4: During scan one, find out the correspondences of large 1-itemset of min FS. And establish a descending data table by the fuzzy value of each transaction data, shown in Table 4.

Step 5: According to the Header table, rebuild fuzzy set of

TABLE IV: Header table

Item	Count
D.Middle	5.333
E.Middle	5.000
F.Middle	4.667
B.Low	4.167
C.Middle	4.167
A.Low	3.833

TABLE V: The new fuzzy sets from Table 2

TID	Fuzzy Sets
1	$\frac{0.833}{D.Middle}, \frac{0.667}{F.Middle}, \frac{0.333}{C.Middle}, \frac{1.000}{A.Low}$
2	$\frac{1.000}{D.Middle}, \frac{0.667}{E.Middle}, \frac{0.500}{F.Middle}, \frac{0.667}{B.Low}, \frac{0.667}{C.Middle}$
3	$\frac{0.667}{E.Middle}, \frac{0.500}{B.Low}, \frac{0}{A.Low}$
4	$\frac{0.833}{D.Middle}, \frac{0.833}{E.Middle}, \frac{0.833}{F.Middle}, \frac{0}{B.Low}, \frac{0.333}{C.Middle}, \frac{0.833}{A.Low}$
5	$\frac{0.500}{E.Middle}, \frac{1.000}{F.Middle}, \frac{0.333}{A.Low}$
6	$\frac{0.833}{D.Middle}, \frac{0}{E.Middle}, \frac{0.667}{C.Middle}, \frac{0.833}{A.Low}$
7	$\frac{1.000}{D.Middle}, \frac{0.167}{E.Middle}, \frac{0.500}{F.Middle}, \frac{0.667}{B.Low}, \frac{0.333}{C.Middle}$
8	$\frac{0.333}{D.Middle}, \frac{1.000}{E.Middle}, \frac{0.667}{F.Middle}, \frac{0}{B.Low}, \frac{0.667}{C.Middle}, \frac{0.500}{A.Low}$
9	$\frac{0.500}{D.Middle}, \frac{0.667}{E.Middle}, \frac{0.500}{F.Middle}, \frac{0.333}{C.Middle}, \frac{0.667}{A.Low}$
10	$\frac{0.833}{E.Middle}, \frac{0}{F.Middle}, \frac{1.000}{B.Low}, \frac{0.333}{C.Middle}, \frac{0}{A.Low}$

the transaction table from Table 2. The new fuzzy set is shown as Table 5.

Step 6: In the second process of scan, establish a fuzzy FP-tree, shown as Fig 5.

Step 7: The item of Header table is mined sequence in descending order, and establish each node of FFP-tree of conditional pattern base. For each conditional pattern base, establish a conditional FFP-tree. Next, repeat the FFP-tree and proceed mining which is ascended to include a frequent pattern of conditional FFP-tree. Finally, if the conditional FFP-tree is contained one particular path, all patterns are listed. The result is shown in Table 6.

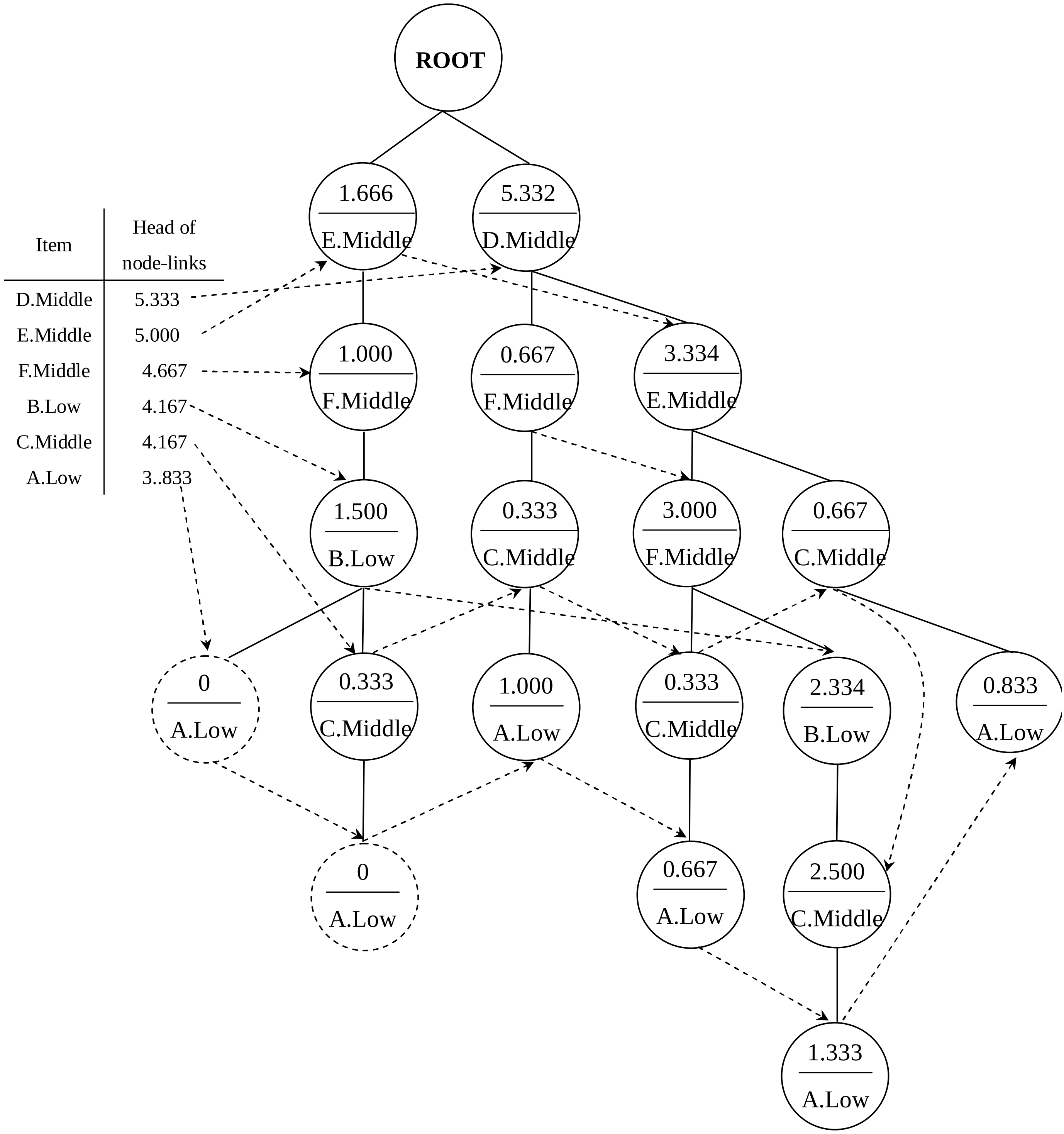

Fig. 5: Fuzzy FP-tree

TABLE VI: The production of frequent patterns

Item	Conditional Pattern-Base	Conditional FFP-Tree	Generate frequent patterns
A.Low	<(E.Middle),(F.Middle),(B.Low)> <(E.Middle),(F.Middle),(B.Low), (C.Middle)> <(D.Middle),(F.Middle),(C.Middle)> <(D.Middle),(E.Middle),(F.Middle), (C.Middle)> <(D.Middle),(E.Middle),(F.Middle),(B.Low), (C.Middle)>	<(D.Middle):4> <(E.Middle):5> <(F.Middle):5> <(B.Low):3>	{(A.Low),(D.Middle)} {(A.Low),(E.Middle)} {(A.Low),(F.Middle)} {(A.Low),(C.Middle)} {(A.Low),(B.Low)} {(A.Low),(D.Middle), (E.Middle),(F.Middle), (C.Middle),(B.Low) }
C.Middle	<(E.Middle),(F.Middle), (B.Low)> <(D.Middle),(F.Middle)> <(D.Middle),(E.Middle)> <(D.Middle),(E.Middle),(F.Middle)> <(D.Middle),(E.Middle),(F.Middle), (B.Low)>	<(D.Middle):4> <(E.Middle):4> <(F.Middle):4> <(B.Low):2>	{(C.Middle),(D.Middle)} {(C.Middle),(E.Middle)} {(C.Middle),(F.Middle)} {(C.Middle),(D.Middle),(E.Middle), (F.Middle)}
B.Low	<(E.Middle), (F.Middle)> <(D.Middle),(E.Middle),(F.Middle)>	<(D.Middle):1> <(E.Middle):2> <(F.Middle):3>	*
F.Middle	<(E.Middle)> <(D.Middle)> <(D.Middle),(E.Middle)>	<(D.Middle):2> <(E.Middle):2>	*
E.Middle	<(D.Middle)>	<(D.Middle):1>	*
D.Middle	*	*	*

Step 8: Using the substeps to mine fuzzy association rules as below.

(a) List all frequent itemsets.

C.Middle $\Rightarrow$ D.Middle
C.Middle $\Rightarrow$ E.Middle
C.Middle $\Rightarrow$ F.Middle
A.Low $\Rightarrow$ D.Middle
A.Low $\Rightarrow$ F.Middle

(b) Count the confidences for all frequent itemsets, if it is greater than or equal to min FC then fuzzy association rules.

C.Middle $\Rightarrow$ D.Middle with
FC (C.Middle $\Rightarrow$ D.Middle) = 0.840:

If a middle number of item C is bought, then a middle number of item D is bought.

C.Middle $\Rightarrow$ F.Middle with
FC (C.Middle $\Rightarrow$ F.Middle) = 0.641:

If a middle number of item C is bought, then a middle number of item F is bought.

C.Middle $\Rightarrow$ E.Middle with
FC (C.Middle $\Rightarrow$ E.Middle) = 0.600:

If a middle number of item C is bought, then a middle number of item E is bought.

A.Low $\Rightarrow$ D.Middle with
FC (A.Low $\Rightarrow$ D.Middle) = 0.869:

If a small number of item A is bought, then a middle number of item D is bought.

A.Low $\Rightarrow$ F.Middle with
FC (A.Low $\Rightarrow$ F.Middle) = 0.652:

If a small number of item A is bought, then a middle number of item F is bought.

The five rules can then serve as meta-knowledge concerning the given transactions.

V. An Experiment Results

The experiment results were proposed method with different database sizes which are set (including 10,000 and 20,000 transaction records generated randomly) and minimum FS, we will discuss the effects caused by execution time (by second) and association rules. We were implemented in Visual Basic 6 and designed Apriori and FFP-growth on an Intel Pentium IV personal computer with 1.70 GHz and 512 MB RAM. In each transaction record, purchased items and numbers were randomly generated. But, purchased numbers cannot exceed ten units and no purchased items should be repeated. The membership functions were shown in Fig. 6. And set minimum FC as 0, we get experiment results shown as table 7.

Table 7 includes the data mined by Apriori and FFP-growth. We find that minimum FC get smaller from table 7, it generates more and more fuzzy association rules. In execution time, the proposed method outdoes Apriori algorithm. It is especially clear when minimum FC gets smaller. It proved that by applying FFP-growth to generate frequent patterns can highly promote the overall efficiency in execution. Furthermore, the smaller minimum FS seemed not to affect the execution time of the method proposed.

VI. Conclusions

The main emphasis of this paper is to propose a fuzzy data mining technique to find fuzzy association rules by using the fuzzy partition method and FP-growth. The features the

TABLE VII: The experiment results

min FS	10,000 transaction records				20,000 transaction records			
	Apriori Algorithm		FFP-growth Algorithm		Apriori Algorithm		FFP-growth Algorithm	
	Execution time	Rule numbers	Execution time	Rule numbers	Execution time	Rule numbers	Execution time	Rule numbers
0.02	612	345	58	243	1342	328	69	239
0.03	583	276	42	175	1183	268	50	189
0.04	323	103	42	167	672	118	50	171
0.05	286	103	22	98	581	105	39	109
0.06	86	103	22	85	189	105	38	92
0.07	21	35	15	61	45	39	31	76
0.08	21	35	11	58	45	35	23	62
0.09	21	35	11	58	45	35	23	62
0.10	21	35	11	58	45	35	23	62

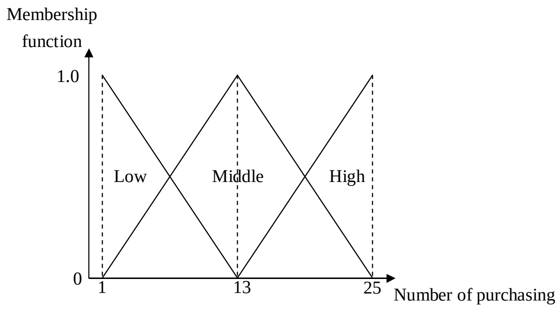

Fig. 6: The membership functions are used in the experiment

proposed method demonstrates are that it does not need to generate candidate itemsets and improves the efficiency of repetitious database scanning. In comparison, the proposed method has achieved better executive efficiency. However, one cannot overlook the advantage of fuzzy association rules in which the natural language is easily used and better comprehended.

Moreover, when it comes to defining linguistic values, managers can select their own preferences, and refer to past experiences and relate cognitive abilities to design a number of linguistic values and shapes, such as Gaussian distribution and trapezoid membership functions. Hence, it would correspond with manager's subjective cognition.

Additionally, the proposed method needs to improve its storage space. In handling the algorithms, the linguistic number θ of each quantitative attribute and recordable number of transaction data are to affect the size of the storage space in the FP-tree. In other words, when θ is bigger, the storage space is bigger; whereas when the recordable number of transaction is bigger, the storage space is bigger. However, saving the number of the linguistic value θ and storage space in the FP-tree will be another issue to ponder. It occurs to us that managing the storage space requires the further discussion in the future.

REFERENCES

[1] Agrawal, R. and Srikant, R., "Fast algorithms for mining association rules," *in Proceedings of 1994 International Conference on Very Large Data Bases*, pp.487-499, 1994.
[2] Berry, M. and Linoff, G., *Data Mining Techniques: for marketing, sales, and customer support*, John Wiley & Sons, NY, 1997.
[3] Chen, S. M., Jong, W. T., "Fuzzy query translation for relational database systems," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 27, no. 4, pp. 714-721, 1997.
[4] Han, J. W. and Kamber, M., *Data Mining : Concepts and Techniques*, Morgan Kaufmann, San Francisco, 2001.
[5] Han, J., Pei, J., and Yin, Y., "Mining Frequent Patterns without Candidate Generation," *in Proc. ACM SIGMOD Int. Conf. on Management of Data*, pp, 1-12, 2000.
[6] Hong, T. P., & Chen, J. B., "Find relevant attributes and membership functions," *Fuzzy Sets and Systems*, Vol. 103, no. 3, pp.389-404, 1999.
[7] Hu Y. C,. "Mining association rules at a concept hierarchy using fuzzy partition," *Journal of Information Management*, Vol. 13, no.3, pp.63-80, 2006.
[8] Hu, Y. C., Chen, R. S. and Tzeng, G. H., "Finding Fuzzy Classification Rules Using Data Mining Techniques," *Pattern Recognition Letters*, vol. 24, pp.509-519, 2003.
[9] Ishibuchi, H., Nakashima, T., and Murata, T. (1999), "Performance Evaluation of Fuzzy Classifier Systems for Multidimensional Pattern Classification Problems," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 29, no.5, pp.601-618, 1999.
[10] Jang, J. S. R., Sun, C. T. and Mizutani, E., *Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence*, Prentic-Hall, NJ, 1997.
[11] Myra, S., "Web usage mining for web site evaluation," *Communications of the ACM*, Vol. 43, pp. 21-30, 1994.
[12] Pedrycz, W., "Why triangular membership functions?," *Fuzzy Sets and Systems*, Vol. 64, pp. 21-30, 1994.
[13] Tan. P. N., Michael Mteinbach, Vipin Kumar, *Introduction to Data Mining*, NY: Addison Wesley, 2005
[14] Wang, L. X. and Mendel, J. M. (1992), "Generating Fuzzy Rules by Learning from Examples", *IEEE Transactions on Systems, Man, and Cybernetisc*, Vol. 22, no.6, pp.1414-1427, 1992.
[15] Zadeh, L. A. (1965), "Fuzzy Sets," *Information Control*, vol. 8, no.3, pp.338-353, 1965.
[16] Zadeh, L. A. (1975), "The Concept of a Linguistic Variable and Its Application to Approximate Reasoning," *Information Science (part 1)*, Vol. 8, no. 3, pp.199-249, 1975.
[17] Zadeh, L. A. (1975), "The Concept of a Linguistic Variable and Its Application to Approximate Reasoning," *Information Science (part 2)*, Vol. 8, no. 4, pp.301-357, 1975.
[18] Zadeh, L. A. (1976), "The Concept of a Linguistic Variable and Its Application to Approximate Reasoning," *Information Science (part 3)*, Vol. 9, no. 1, pp.43-80, 1976.
[19] Zimmermann, H. -J., *Fuzzy sets, Decision making and expert systems*, Kluwer, Boston, 1991.