Finding Local and Periodic Association Rules from Fuzzy Temporal Data

F. A. Mazarbhuiya, M. Shenify, Md. Husamuddin
College of Computer Science and IT
Albaha University, Albaha, KSA
fokrul_2005@yahoo.com
mshenify@yahoo.com
mdhusamuddin@gmail.com

Abstract. The problem of finding association rules from a dataset is to find all possible associations that hold among the items, given a minimum support value and a minimum confidence. This involves finding frequent sets first and then the association rules that hold within the items in the frequent sets. The problem of mining temporal association rules from temporal dataset is to find association rules between items that hold within certain time intervals but not throughout the dataset. This involves finding frequent sets that are frequent at certain time intervals and then association rules among the items present in the frequent sets. In some of the applications the time of transaction is imprecise; we call the associated dataset as fuzzy temporal dataset. In such datasets, we may find set of items that are frequent in certain fuzzy time intervals. We call these as locally frequent sets over fuzzy time intervals and the associated association rules as local association rules over fuzzy time intervals. These association rules cannot be discovered in the usual way because of fuzziness involved in temporal features. Normally these association rules are periodic in nature. We call such rules as periodic association rules over fuzzy time interval. We propose modification to the A-priori algorithm to compute locally frequent sets and to extract periodic frequent sets and periodic association rules from fuzzy temporal data.

Key-words: Core of a fuzzy number, Data mining, frequent sets, fuzzy membership function, $\alpha$-cut.

1 Introduction

The problem of mining association rules has been defined initially [15] by R. Agarwal et al for application in large super markets. Large supermarkets have large collection of records of daily sales. Analyzing the buying patterns of the buyers will help in taking typical business decisions such as what to put on sale, how to put the materials on the shelves, how to plan for future purchase etc.

Mining for association rules between items in temporal databases has been described as an important data-mining problem. Transaction data are normally temporal. The market basket transaction is an example of this type.

In this paper we consider datasets, which are fuzzy temporal i.e. the time in which a transaction has taken place is imprecise or approximate and is attached to the transactions. In large volumes of such data, some hidden information or relationship among the items may be there which cannot be extracted because of some fuzziness in the temporal features. Also the case may be that some association rules may hold in certain fuzzy time period but not throughout the dataset. For finding such association rules we need to find itemsets that are frequent at certain time period, which will obviously be imprecise due to the fact that the time of each transaction is fuzzy. We call such frequent sets locally frequent over fuzzy time interval. From these locally frequent sets, associations among the items in these sets can be obtained. Since a periodic nature is there in any natural event this kind of associations normally hold periodically. And if such locally frequent sets also have the property that they become frequent in certain fuzzy time intervals i.e.
they are periodic in nature then we call these sets periodic frequent sets and the associated associations as periodic association rules over fuzzy time interval.

In section 2 we give a brief discussion on the recent works in Temporal Data Mining and fuzzy temporal data mining. In section 3 we describe the terms and notations used in this paper. In section 4, we give the algorithm proposed in this paper for mining locally frequent sets over fuzzy time interval and local association rules over same. In section 5, we discuss about periodic association rule over fuzzy time interval. We conclude with conclusion and lines for future work in section 6.

2 Recent works

The problem of discovery of association rules was first formulated by Agrawal et al. in 1993. Given a set \( I \), of items and a large collection \( D \) of transactions involving the items, the problem is to find relationships among the items i.e. the presence of various items in the transactions. A transaction \( t \) is said to support an item if that item is present in \( t \). A transaction \( t \) is said to support an itemset if \( t \) supports each of the items present in the itemset. An association rule is an expression of the form \( X \Rightarrow Y \) where \( X \) and \( Y \) are subsets of the itemset \( I \). The rule holds with confidence \( \tau \) if \( \tau \% \) of the transactions in \( D \) that supports \( X \) also supports \( Y \). The rule has support \( \sigma \) if \( \sigma \% \) of the transactions supports \( X \cup Y \). A method for the discovery of association rules was given in [15], which is known as the A priori algorithm. This was then followed by subsequent refinements, generalizations, extensions and improvements. As the number of association rules generated is too large, attempts were made to extract the useful rules ([13], [16]) from the large set of discovered association rules. Attempts are also made to make the process of discovery of rules faster ([12], [14]). Generalized association rules ([9], [17]) and Quantitative association rules ([18]) were later on defined and algorithms were developed for the discovery of these rules. A hashed based technique is used in [11] to improve the rule mining process of the A priori algorithm.

Temporal Data Mining is now an important extension of conventional data mining and has recently been able to attract more people to work in this area. By taking into account the time aspect, more interesting patterns that are time dependent can be extracted. There are mainly two broad directions of temporal data mining [7]. One concerns the discovery of causal relationships among temporally oriented events. Ordered events form sequences and the cause of an event always occur before it. The other concerns the discovery of similar patterns within the same time sequence or among different time sequences. The underlying problem is to find frequent sequential patterns in the temporal databases. The name sequence mining is normally used for the underlying problem. In [8] the problem of recognizing frequent episodes in an event sequence is discussed where an episode is defined as a collection of events that occur during time intervals of a specific size.

The association rule discovery process is also extended to incorporate temporal aspects. In temporal association rules each rule has associated with it a time interval in which the rule holds. The problems associated are to find valid time periods during which association rules hold, the discovery of possible periodicities that association rules have and the discovery of association rules with temporal features. In [10], [19], [20] and [21], the problem of temporal data mining is addressed and techniques and algorithms have been developed for this. In [10] an algorithm for the discovery of temporal association rules is described. In [2], two algorithms are proposed for the discovery of temporal rules that display regular cyclic variations where the time interval is specified by user to divide the data into disjoint segments like months, weeks, days etc.

Similar works were done in [6] and [22] incorporating multiple granularities of time intervals (e.g. first working day of every month) from which both cyclic and user defined calendar patterns can be achieved. In [1], the method of finding locally and periodically frequent sets and periodic association rules are discussed which is an improvement of other methods in the sense that it dynamically extract all the rules along with the intervals where the rules hold. In ([23], [24]) fuzzy calendric data mining and fuzzy temporal data mining is discussed where user specified ill-defined
3.1 Some Definitions related to Fuzziness

Let $E$ be the universe of discourse. A fuzzy set $A$ in $E$ is characterized by a membership function $A(x)$ lying in $[0, 1]$. $A(x)$ for $x \in E$ represents the grade of membership of $x$ in $A$. Thus a fuzzy set $A$ is defined as

$$A = \{(x, A(x)), x \in E\}$$

A fuzzy set $A$ is said to be normal if $A(x) = 1$ for at least one $x \in E$

An $\alpha$-cut of a fuzzy set is an ordinary set of elements with membership grade greater than or equal to a threshold $\alpha$, $0 \leq \alpha \leq 1$. Thus a $\alpha$-cut $A_\alpha$ of a fuzzy set $A$ is characterized by

$$A_\alpha = \{x \in E; A(x) \geq \alpha\} \text{ [see e.g. [4]]}$$

A fuzzy set is said to be convex if all its $\alpha$-cuts are convex sets [see e.g. [5]].

A fuzzy number is a convex normalized fuzzy set $A$ defined on the real line $R$ such that

1. there exists an $x_0 \in R$ such that $A(x_0) = 1$, and
2. $A(x)$ is piecewise continuous.

A fuzzy number is denoted by $[a, b, c]$ with $a < b < c$ where $A(a) = A(c) = 0$ and $A(b) = 1$. $A(x)$ for all $x \in [a, b]$ is known as left reference function and $A(x)$ for $x \in [b, c]$ is known as the right reference function. Thus a fuzzy number can be thought of as containing the real numbers within some interval to varying degrees. The $\alpha$-cut of the fuzzy number $[t_1-a, t_1, t_1+a]$ is a closed interval $[t_1+(-\alpha1).a, t_1+(1-\alpha1).a]$.

Fuzzy intervals are special fuzzy numbers satisfying the following.

1. there exists an interval $[a, b] \subset R$ such that $A(x_0) = 1$ for all $x_0 \in [a, b]$, and
2. $A(x)$ is piecewise continuous.

A fuzzy interval can be thought of as a fuzzy number with a flat region. A fuzzy interval $A$ is denoted by $A = [a, b, c, d]$ with $a < b < c < d$ where $A(a) = A(d) = 0$ and $A(x) = 1$ for all $x \in [b, c]$. $A(x)$ for all $x \in [a, b]$ is known as left reference function and $A(x)$ for $x \in [c, d]$ is known as the right reference function. The left reference function is non-decreasing and the right reference function is non-increasing [see e.g. [3]].

Similarly the $\alpha$-cut of the fuzzy interval $[t_1-a, t_1, t_2, t_2+a]$ is a closed interval $[t_1+(-\alpha1).a, t_2+(1-\alpha1).a]$.

The core of a fuzzy number $A$ is the set of elements of $A$ having membership value one i.e.

$$\text{Core}(A) = \{(x, A(x)); A(x) = 1\}$$

For every fuzzy set $A$, fuzzy temporal and calendric patterns are extracted from temporal data.

Our approach is different from the above approaches. We are considering the fact that the time of transactions are not precise rather they are fuzzy numbers and some items are seasonal or appear frequently in the transactions for certain ill-defined periods only i.e. summer, winter, etc. They appear in the transactions for a short time and then disappear for a long time. After this they may again reappear for a certain period and this process may repeat. For these items the support cannot be calculated in the usual way ([1], [10]), it has to be computed by the method defined in section 3.2. These items may lead to interesting association rules over fuzzy time intervals. In this paper we calculate the support values of these sets locally in a fuzzy time interval. The large fuzzy time gap in which they do not appear is not counted. We also define periodic frequent sets and periodic association rules over fuzzy time intervals. As mentioned in the previous paragraph similarly works were also done in [23], [24] but in non-fuzzy temporal data. But in all these methods they discuss the association rule mining of non-fuzzy temporal data. Our approach although little bit similar to the work of [1], is different from others in the sense that it discovers association rules from fuzzy temporal data and finds the association rules along with their fuzzy time intervals over which the rules hold automatically.

3 Terms, Notations and Symbols used

3.1 Some Definitions related to Fuzziness

Let $E$ be the universe of discourse. A fuzzy set $A$ in $E$ is characterized by a membership function $A(x)$ lying in $[0, 1]$. $A(x)$ for $x \in E$ represents the grade of membership of $x$ in $A$. Thus a fuzzy set $A$ is defined as

$$A = \{(x, A(x)), x \in E\}$$

A fuzzy set $A$ is said to be normal if $A(x) = 1$ for at least one $x \in E$

An $\alpha$-cut of a fuzzy set is an ordinary set of elements with membership grade greater than or equal to a threshold $\alpha$, $0 \leq \alpha \leq 1$. Thus a $\alpha$-cut $A_\alpha$ of a fuzzy set $A$ is characterized by

$$A_\alpha = \{x \in E; A(x) \geq \alpha\} \text{ [see e.g. [4]]}$$

A fuzzy set is said to be convex if all its $\alpha$-cuts are convex sets [see e.g. [5]].

A fuzzy number is a convex normalized fuzzy set $A$ defined on the real line $R$ such that

1. there exists an $x_0 \in R$ such that $A(x_0) = 1$, and
2. $A(x)$ is piecewise continuous.

A fuzzy number is denoted by $[a, b, c]$ with $a < b < c$ where $A(a) = A(c) = 0$ and $A(b) = 1$. $A(x)$ for all $x \in [a, b]$ is known as left reference function and $A(x)$ for $x \in [b, c]$ is known as the right reference function. Thus a fuzzy number can be thought of as containing the real numbers within some interval to varying degrees. The $\alpha$-cut of the fuzzy number $[t_1-a, t_1, t_1+a]$ is a closed interval $[t_1+(-\alpha1).a, t_1+(1-\alpha1).a]$.

Fuzzy intervals are special fuzzy numbers satisfying the following.

1. there exists an interval $[a, b] \subset R$ such that $A(x_0) = 1$ for all $x_0 \in [a, b]$, and
2. $A(x)$ is piecewise continuous.

A fuzzy interval can be thought of as a fuzzy number with a flat region. A fuzzy interval $A$ is denoted by $A = [a, b, c, d]$ with $a < b < c < d$ where $A(a) = A(d) = 0$ and $A(x) = 1$ for all $x \in [b, c]$. $A(x)$ for all $x \in [a, b]$ is known as left reference function and $A(x)$ for $x \in [c, d]$ is known as the right reference function. The left reference function is non-decreasing and the right reference function is non-increasing [see e.g. [3]].

Similarly the $\alpha$-cut of the fuzzy interval $[t_1-a, t_1, t_2, t_2+a]$ is a closed interval $[t_1+(-\alpha1).a, t_2+(1-\alpha1).a]$.

The core of a fuzzy number $A$ is the set of elements of $A$ having membership value one i.e.

$$\text{Core}(A) = \{(x, A(x)); A(x) = 1\}$$

For every fuzzy set $A$,
Obviously for any two fuzzy numbers having similar membership functions right reference of place. We assume that of each other if the slope of the left reference function is equal to the that of B(x) and the slope of right reference of A(x) is equal that of B(x). Obviously for any two fuzzy numbers A and B having similar membership functions

\[ |A| = |B|, \forall a \in [0, 1] \]

### 3.2 Some Definitions related to Association Rule Mining over Fuzzy time period

Let T = \{t_0, t_1, ..., \} be a sequence of imprecise or fuzzy time stamps over which a linear ordering \(<\) is defined where \(t_i < t_j\) means \(t_i\) denotes the core of a fuzzy time which is earlier than the core of another fuzzy time stamp \(t_j\). For the sake of convenience, we assume that all the fuzzy time stamps are having similar membership functions. Let I denote a finite set of items and the transaction database \(D\) is a collection of transactions where each transaction has a part which is a subset of the itemset \(I\) and the other part is a fuzzy time-stamp indicating the approximate time in which the transaction had taken place. We assume that \(D\) is ordered in the ascending order of the core of fuzzy time stamps. For fuzzy time intervals we always consider a fuzzy closed intervals of the form \([t_1-\alpha, t_2, t_2+\alpha]\) for some real number \(\alpha\). We say that a transaction is in the fuzzy time interval \([t_1-\alpha, t_2, t_2+\alpha]\) if the \(\alpha\)-cut of the fuzzy time stamp of the transaction is contained in \(\alpha\)-cut of \([t_1-\alpha, t_2, t_2+\alpha]\) for some user’s specified value of \(\alpha\).

We define the local support of an itemset in a fuzzy time interval \([t_1-\alpha, t_2, t_2+\alpha]\) as the ratio of the number of transactions in the time interval \([t_1+(\alpha-1), a, t_2+1(1-\alpha), a]\) containing the itemset to the total number of transactions in \([t_1+(\alpha-1), a, t_2+1(1-\alpha), a]\) for the whole data base \(D\) for a given value of \(\alpha\). We use the notation \(\text{Sup}_{[t_1-\alpha, t_2, t_2+\alpha]}(X)\) to denote the support of the itemset \(X\) in the fuzzy time interval \([t_1-\alpha, t_1, t_2, t_2+\alpha]\). Given a threshold \(\sigma\) we say that an itemset \(X\) is frequent in the fuzzy time interval \([t_1-\alpha, t_1, t_2, t_2+\alpha]\). We say that an association rule \(X \Rightarrow Y\), where \(X\) and \(Y\) are item sets holds in the time interval \([t_1-\alpha, t_1, t_2, t_2+\alpha]\) if and only if given threshold \(\tau\),

\[ \text{Sup}_{[t_1-\alpha, t_1, t_2+\alpha]}(X \cup Y)/\text{Sup}_{[t_1-\alpha, t_1, t_2+\alpha]}(X) \geq \tau\]

and \(X \cup Y\) is frequent in \([t_1-\alpha, t_1, t_2, t_2+\alpha]\). In this case we say that the confidence of the rule is \(\tau\).

For each locally frequent item set we keep a list of fuzzy time intervals in which the set is frequent where each fuzzy interval is represented as \([start-a, start, end, end+a]\) where \(start\) gives the approximate starting time of the time interval and \(end\) gives the approximate ending time of the time-interval. \(end - start\) gives the length of the core of the fuzzy time interval. For a given value of \(\alpha\) of two intervals \([start_1-a, start_1, end_1, end_1+a]\) and \([start_2-a, start_2, end_2, end_2+a]\) are non-overlapping if their \(\alpha\)-cuts are non-overlapping.

### 4 Algorithm proposed:

#### 4.1 Generating Locally Frequent Sets

While constructing locally frequent sets, with each locally frequent set a list of fuzzy time-intervals is maintained in which the set is frequent. Two user’s specified thresholds \(\alpha\) and minsup are used for this. During the execution of the algorithm while making a pass through the database, if for a particular itemset the \(\alpha\)-cut of its current fuzzy time-stamp, \([\text{Current}, \text{RCurrent}]\) and the \(\alpha\)-cut, \([\text{RLastseen}, \text{RLastseen}]\) of its fuzzy time, when it was last seen overlap then the current transaction is included in the current time-interval under consideration which is extended with replacement of \(\text{RLastseen}\) by \(\text{RCurrent}\); otherwise a new time-interval is started with \(\text{LCurrent}\) as the starting point. The support count of the item set in the previous time interval is checked to see whether it is frequent in that interval.
or not and if it is so then it is fuzzified and added to the list maintained for that set. Also for the locally frequent sets over fuzzy time intervals, a minimum core length of the fuzzy period is given by the user as \(\text{minthd}\) and fuzzy time intervals of core length greater than or equal to this value are only kept. If \(\text{minthd}\) is not used than an item appearing once in the whole database will also become locally frequent over fuzzy point of time.

Procedure to compute \(L_1\), the set of all locally frequent item sets of size 1.

For each item while going through the database we always keeps an \(\alpha\)-cut \(\alpha_{\text{lastseen}}\) which is \([\alpha_{L_{\text{lastseen}}}, \alpha_{R_{\text{lastseen}}}]\) that corresponds to the fuzzy time stamp when the item was last seen. When an item is found in a transaction and the fuzzy time-stamp is \(tm\) and if its \(\alpha\)-cut \(\alpha_{tm} = [\alpha_{Ltm}, \alpha_{Rtm}]\) has empty intersection with \([\alpha_{L_{\text{lastseen}}}, \alpha_{R_{\text{lastseen}}}]\), then a new time interval is started by setting start of the new time interval as \(\alpha_{Ltm}\) and end of the previous time interval as \(\alpha_{R_{\text{lastseen}}}\). The previous time interval is fuzzified provided the support of the item is greater than \(\text{min-sup}\). The fuzzified interval is then added to the list maintained for that item provided that the duration of the core is greater than \(\text{minthd}\). Otherwise \(\alpha_{R_{\text{lastseen}}}\) is set to \(\alpha_{Rtm}\), the counters maintained for counting transactions are increased appropriately and the process is continued.

Following is the algorithm to compute \(L_1\), the list of locally frequent sets of size 1. Suppose the number of items in the dataset under consideration is \(n\) and we assume an ordering among the items.

**Algorithm 4.1**

\[
C_1 = \{(i_k, tp[k]) : k = 1, 2, \ldots, n\}
\]

where \(i_k\) is the \(k\)-th item and \(tp[k]\) points to a list of fuzzy time intervals initially empty.

for \(k = 1\) to \(n\) do

set \(\text{lastseen}[k] = \phi\);

set \(\text{itemcount}[k]\) and \(\text{transcount}[k]\) to zero for each transaction \(t\) in the database with fuzzy time stamp \(tm\) do

\{for \(k = 1\) to \(n\) do

\{ if \(\{i_k\} \subseteq t\) then

\{ if \(\text{lastseen}[k] = \phi\)

\{ \(\text{lastseen}[k] = \text{firstseen}[k] = tm; \)
\}

\} else

if \((\text{lastseen}[k], \text{Rlastseen}[k]) \cap

[\text{Ltm}[k], \text{Rtm}[k]] = \phi\)

\{ \(\text{Rlastseen}[k] = \text{Rtm}[k]; \)
\itemcount[k]++;\}

else

\{ if \(\itemcount[k]/\text{transcount}[k] \times 100 \geq \alpha\)

fuzzify\((\text{lastseen}[k], \text{Rlastseen}[k]), \forall \alpha \in [0, 1]\)

if \((|\text{core}(\text{fuzzified interval})| \geq \text{minthd})\)

add(\text{fuzzified interval}) to \(tp[k]\);

\itemcount[k] = \text{transcount}[k] = 1;

\text{lastseen}[k] = \text{firstseen}[k] = tm;

\}

\}

\} // end of k-loop //

// end of do loop //

for \(k = 1\) to \(n\) do

\{ if \(\itemcount[k]/\text{transcount}[k] \times 100 \geq \alpha\)

fuzzify\((\text{lastseen}[k], \text{Rlastseen}[k]), \forall \alpha \in [0, 1]\)

if \((|\text{core}(\text{fuzzified interval})| \geq \text{minthd})\)

add(\text{fuzzified interval}) to \(tp[k]\);

if \(\text{tp}[k] \neq 0\) add \{\(i_k, tp[k]\}\) to \(L_1\)

\}

fuzzify\((a, b), \alpha\)

\{ \(\text{fuzzified interval} = \bigcup_{\alpha \in [0, 1]} [a, b];\)

where \([a, b](x) = \alpha \cdot [a, b](x)\)

return(\text{fuzzified interval})
\}

Two support counts are kept, \(\text{itemcount}\) and \(\text{transcount}\). If the count percentage of an item in an \(\alpha\)-cut of a fuzzy time interval is greater than the minimum threshold then only the set is considered as a locally frequent set over fuzzy time interval.

\(L_1\) as computed above will contain all 1-sized locally frequent sets over fuzzy time intervals and with each set there is associated an ordered list of fuzzy time intervals in which the set is frequent. Then A priori candidate generation algorithm is
used to find candidate frequent set of size 2. With each candidate frequent set of size two we associate a list of fuzzy time intervals that are obtained in the pruning phase. In the generation phase this list is empty. If all subsets of a candidate set are found in the previous level then this set is constructed. The process is that when the first subset appearing in the previous level is found then that list is taken as the list of fuzzy time intervals associated with the set. When subsequent subsets are found then the list is reconstructed by taking all possible pair wise intersection of subsets one from each list. Sets for which this list is empty are further pruned.

Using this concept we describe below the modified A-priori algorithm for the problem under consideration.

**Algorithm 4.2**

**Modified A priori**

Initialize

\[ k = 1; \]
\[ C_1 = \text{all item sets of size 1} \]
\[ L_1 = \{ \text{frequent item sets of size 1 where with each itemset } \{i, k\} \text{ a list } tp[i]\text{ is maintained which gives all time fuzzy intervals in which the set is frequent} \} \]

\[ L_1 \text{ is computed using algorithm 1.1 } /*\]

for \( k = 2; L_{k-1} \neq \emptyset; k++ \) do

\[ C_k = \text{apriori-gen}(L_{k-1}) /* \text{same as the candidate generation method of the A priori algorithm setting } tp[i] \text{ to zero for all } i *\]

\[ \text{prune}(C_k); \]

\[ \text{drop all lists of fuzzy time intervals maintained with the sets in } C_k \]

\[ \text{Compute } L_k \text{ from } C_k; \]

\[ //L_k \text{ can be computed from } C_k \text{ using the same procedure used for computing } L_1 // \]

\[ k = k + 1 \]

Answer = \( \bigcup_k L_k \)

Prune(C_k)

\{Let \( m \) be the number of sets in \( C_k \) and the sets be \( s_1, s_2, ..., s_m\) Initialize the pointers \( tp[i] \) pointing to the list of fuzzy time intervals maintained with each set \( s_i \) to null

for \( i = 1 \) to \( m \) do

{for each \((k-1)\) subset \( d \) of \( s_i \) do

{if \( d \notin L_{k-1} \) then

\[ C_k = C_k - \{s_i, tp[i]\}; \text{break;} \]

else

\{if \( tp[i] = \text{null} \) then set \( tp[i] \) to point to the list of fuzzy time intervals maintained for \( d \)

\{take all possible pair-wise intersection of fuzzy time intervals one from each list, one list maintained with \( tp[i] \) and the other maintained with \( d \) and take this as the list for \( tp[i] \)

\[ \text{delete all fuzzy time intervals whose core length is less than the value of minthd if } tp[i] \text{ is empty then } C_k = C_k - \{s_i, tp[i]\}; \text{break; } \]

\}

\}

\}

}\}

\}

4.2 Generating Association Rules

If an itemset is frequent in a fuzzy time-interval \([t_1-a, t_1, t_2, t_2+a]\) then all its subsets are also frequent in the fuzzy time-interval \([t_1-a, t_1, t_2, t_2+a]\). But to generate the association rules as defined in section 3, we need the supports of the subsets in fuzzy time-interval \([t_1-a, t_1, t_2, t_2+a]\), which may not be available after application of the algorithm as defined in 4.1. For this one more scan of the whole database will be needed. For each association rule we attach a fuzzy interval in which the association rule holds.

5 Extracting Periodic Patterns

In the algorithm proposed in section 4 for each locally frequent set, a list of fuzzy time intervals is maintained in which the set is frequent. In this list if we find that the distance between the cores of fuzzy intervals is almost equal (up to a small variation) then we call these frequent sets periodic frequent sets over fuzzy time intervals. Association rules that hold periodically over fuzzy time period are called periodic association rules over fuzzy time interval.
6 Conclusion and Lines for future work

An algorithm for finding frequent sets that are frequent in certain fuzzy time periods from fuzzy temporal data, in the paper is given. The algorithm dynamically computes the frequent sets along with their fuzzy time intervals where the sets are frequent. These frequent sets are named as locally frequent sets over fuzzy time interval. The technique used is similar to the A priori algorithm. From these locally frequent sets interesting rules may follow. Some of these locally frequent sets may be periodic in nature. Then we call these sets periodic frequent sets over fuzzy time interval and the associated rules periodic association rule over fuzzy time interval.

In the level-wise generation of locally frequent sets, for each locally frequent set we keep a list of all fuzzy time-intervals in which it is frequent. For generating candidates for the next level, pair-wise intersection of the intervals in two lists are taken. The same algorithm can be implemented with both real life as well as synthetic datasets.

REFERENCES


[24] W. J. Lee and S. J. Lee; Fuzzy Calendar Algebra and It's Applications to Data Mining, Proceedings of the 11th International