International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868
Foundation of Computer Science FCS, New York, USA
Volume 9 – No.2, June 2015 – www.ijais.org

An Efficient Method for Generating Local Association Rules

F. A. Mazarbhuiya, Ph.D
Assistant Professor
Department of Computer Science & IT.
Al Baha University, Al Baha,
Kingdom of Saudi Arabia (KSA)

Yusuf Perwej, Ph.D
Ph.D, M.Tech (Computer Science & Engg.)
Assistant Professor
Department of Computer Science & Engg.
Al Baha University, Al Baha,
Kingdom of Saudi Arabia (KSA)

ABSTRACT
Carving association rules from any available set is a pre-defined problem and there are a variety of methods available for the extraction of association rules. Almost in all the cases the major emphasis is given to the generating most occurring itemsets rather than the extraction of association rules. Only a few numbers of researchers have through some light on this problem. Extracting generally most occurring itemsets for sometimes does not guarantee that particular dataset will occur in its lifetime. . For finding local association rules of the form A $\Rightarrow$ X - A, where X and A are itemsets that hold in the interval [t, t′] and $A \subset X$. In order to calculate the confidence of the rule A $\Rightarrow$ X – A in the interval [t, t′], it is required to know the supports of both X and A in the same interval [t, t′]. The supports of X and any of its subset A may not be available for the same time interval - A may be frequent in an interval greater than [t, t′]. So, they have loosely defined local association rule as confidence in the rule A $\Rightarrow$ X – A can't be calculated in interval [t, t′]. In this paper, we present a latest approach for finding association rules from generally most occurring item sets using Rough Set and Boolean reasoning. The rules carved are called local association rules. The efficacy of the proposed approach is established through experiment over retail dataset that contains retail market basket data from an anonymous Belgian retail store.

Keywords
Data Mining, Temporal Data Mining, Local Association Rule Mining, Rough Set, Boolean Reasoning

1. INTRODUCTION
Mining association rules in transaction data are a so much explored problem in the field of data mining. The problem is to extract all rules of the type – "what goes with what". The problem was formulated by Agrawal et al. [1, 2] in 1993. In this task, a set of items and a huge set of transactions are provided; the problem is to deduce the relationships among items that are provided to a user's support and confidence satisfaction level. However, the transaction data are temporal in the sense that when a transaction happens the time of transaction is also recorded. Considering the time aspect, different methods [3] have been proposed to extract temporal association rules, i.e., rules that hold throughout the lifetime of the itemset rather than throughout the lifetime of the dataset. The lifetime of an itemset is the time period between the first transaction and the last transaction containing the same itemset in the dataset and it may not be same as the lifetime of the dataset. Similar to general association rule mining problem, temporal association rule carving is defined as a two-step process (i) finding most occurring itemsets which are frequent throughout their lifetime, and (ii) generating association rules that hold throughout the lifetime of the most occurring itemsets. In most of the existing work [4] emphasis is given to finding most occurring itemsets rather than the extraction of association rules from temporal data sets. Anjana et al. Have addressed the problem of temporal association rule extraction in [8]. They proposed an algorithm for finding itemsets with respect to a small time-period, not necessarily equal to the lifetime of the data set or that of the itemset. They named such itemsets as locally frequent itemsets and corresponding rules as local association rules. In order to calculate the confidence value of a local association rule, say A $\Rightarrow$X – A, in the interval $[t,t']$where X is a frequent a itemset in $[t,t']$and $A \subset X$, it is required to know the supports of both X and A in the same interval $[t,t']$. But, the way supports of itemsets are calculated in [8], the support of subsets of X may not be available for the same time interval rather they may be frequent in an interval greater than $[t,t']$. So, they have loosely defined association rules, as confidence of the rule A $\Rightarrow$ X – A cannot be calculated in interval $[t,t']$directly. Rough set theory, proposed by Pawlak [13], seems to be a solution to this problem. Nguyen et al. [14] have presented a method of extracting association rules, based on rough set and Boolean reasoning. They have shown a relationship between association rule mining problem and reducts finding problem in rough set theory. But, their works were mainly focused on non-temporal datasets. In this paper, we present a novel approach for finding local association rules from locally frequent itemsets using rough set and Boolean reasoning. For a given locally frequent itemset X in the time interval $[t,t']$, all those transactions generated between t and t′ are considered and mapped to decision table in line with [14]. Thereafter, we find the reducts using rough set theory and Boolean reasoning to generate association rules that would be local to the time interval $[t,t']$. The efficiency of the proposed approach is established through experiment over retail dataset containing retail market basket data from an anonymous Belgian retail store available on the internet. The rest of the paper is organized as follows: Section 2 presents the related works on temporal association rule mining. Basic concepts, definitions and notations are presented in section 3. The proposed local association rule mining method using rough set and Boolean reasoning is described in section 4. The experimental setup is presented in section 5. Finally, we conclude the paper with possible future directives in section 6.

2. RELATED WORK

Association rules discovery problem was first formulated by Agrawal el al. [1] In 1993. An algorithm for finding association rules was given in [2], which is called as A-priori algorithm. Then there were subsequent refinements, generalizations, extensions and improvements of the problems. The attempts were also made to carve useful rules from large set of association rules [10, 11]. An extension of the above problem is Temporal Data Mining [5]. By taking time attribute into accounts, time-dependent patterns can be extracted**.** There are mainly two broad directions of the above-mentioned problem**.** One concern about the discovery of causal relationships among temporally oriented events and the other about finding of similar patterns within the same time sequence or among different time sequences**.** In [9], the author discussed about the problem of recognizing frequent episodes. In [4], [5], [6], and [12] the authors put forwarded the techniques of discovering patterns from temporal data**.** In 2001 Ale at al [3], developed an algorithm for the discovery of temporal association. For every item (which extends to item set) a lifetime or lifespan is defined which is the time gap between the first occurrence and the last occurrence of the item in the transaction database. The items are calculated only during its lifespan. Thus every rule has allied with it a time frame corresponding to the lifetime of the items coalesce in the rule. In [8], the works done in [3] have been extended by considering time gap between two consecutive transactions containing an item set into account. The frequent itemsets extracted by above method are termed as locally frequent itemsets. Although the method proposed in [3], [8] can extract more frequent itemsets than others; the methods did not address association rules extraction problem adequately. In [15], the author proposed a method of extracting temporal association rules from locally frequent itemsets. They termed the rules as local association rules and are as follows: Suppose X is locally frequent in $[t_1, t_2]$ then its support is from the algorithm proposed in [8]. A being subset of X will also be locally frequent in $[t_1, t_2]$ but A may be locally frequent in a larger interval, say $[t_1', t_2']$ that properly contains $[t_1, t_2]$. So it is clear that in the time periods $[t_1', t_1]$ and $[t_2, t_2']$, X does not occur frequently i.e. X occurs rarely. So in calculating the confidence of the rule $A \Rightarrow X - A$, they consider the ratio $\sup_{[t_1,t_2]}(X)/\sup_{[t_1',t_2']}(A)$ instead of $\sup_{[t_1',t_2']}(X)/\sup_{[t_1',t_2']}(A)$ and lower down the minimum confidence value a bit to extract all confident rules. The procedure is repeated for every set-subset pair. Such a rule $A \Rightarrow X - A$ can be interpreted as that within a intensive region of A in $[t_1', t_2']$ there is a intensive region of the X in $[t_1, t_2]$ where $[t_1, t_2]$ is a subinterval of $[t_1', t_2']$.

The rough sets theory proposed by Pawlak [13] along with several data mining tasks, namely pattern extraction, discretization etc. has been found a useful tool for data mining. In this paper by using this theory a rule that is similar to maximal association rules can be extracted. Nguyen et al. [14], showed the relationship between the problem of association rule extraction for transaction data and relative reducts for a decision table in rough set theory. Alike works have been proposed in [16, 17, 18]. But, in most of the works temporal attribute which is naturally available in a transaction dataset is not taken into consideration. Finally in this paper, we present a noble method for extracting association rules over temporal data set using rough set theory.

3. DEFINITION AND NOTATION USED

Let T = $<t_o, t_1, \ldots\ldots\ldots\ldots>$ be a sequence of timestamps over which a linear ordering < is defined where $t_i < t_j$ means t_i denotes a time which is earlier than t_j. Let I denote a finite set of items and the transaction dataset D is a collection of transactions where each transaction has a part which is a subset of the item set I and the other part is a timestamp indicating the time in which the transaction had taken place. We assume that D orders in the ascending order of the timestamps. For time intervals we always consider closed intervals of the form $[t_1, t_2]$ where t_1 and t_2 are timestamps. We say that a transaction is in the time interval $[t_1, t_2]$ if the time-stamp of the transaction say t is such that $t_1 \le t \le t_2$.

We define the local support of an item set in a time interval $[t_1, t_2]$ as the ratio of the number of transactions in the time interval $[t_1, t_2]$ containing the item set to the total number of transactions in $[t_1, t_2]$ for the whole dataset D. We use the notation $Supp_{[t_1,t_2]}(X)$ to denote the support of the item set X in the time interval $[t_1, t_2]$. Given a threshold σ we say that an item set X is frequent in the time interval $[t_1, t_2]$ if $Supp_{[t_1,t_2]}(X) \ge (\sigma/100)*$ tc where tc denotes the total number of transactions in D that are in the time interval $[t_1, t_2]$. We say that an association rule $X \Rightarrow Y$, where X and Y are item sets holds in the time interval $[t_1, t_2]$ if and only if given threshold τ,

$$Supp_{[t_1,t_2]}(X \cup Y) / Supp_{[t_1,t_2]}(X) \ge \tau/100.0$$

and X∪Y is frequent in $[t_1, t_2]$. In this case we say that the confidence of the rule is τ.

3.1 Algorithm for finding local association rules

For finding an association rule of the form $A \Rightarrow X\text{-}A$ where X and A are item sets that holds in a time interval [t, t'] we need to know the supports of X and A in [t, t']. But the way in which supports of item sets are calculated in the algorithm proposed above the supports of X and any of its subsets A may not be available for the same time interval [t, t']. Suppose X is locally frequent in [t, t'] then its support in [t, t'] is known from the algorithm. A being a subset of X will also be locally frequent in [t, t'] but A may be locally frequent in a larger interval that contains [t, t'] properly. Then the local support of A will be known for the larger interval only. Thus to know the support of an item set and all its subsets in the same time interval we need to make several passes through the dataset keeping several counters for each item set for each of the intervals in which it is locally frequent. This really will be an expensive operation. In this situation we propose to calculate association rule in the following way. Suppose a set X is locally frequent in $t_X = [t_1, t_2]$ and $A \subseteq X$. Then A definitely will be locally frequent in some interval $t_A = [t_1', t_2']$ where t_X is included in t_A. The algorithm proposed in section-3.3 for finding locally frequent item sets will give in its output support of X in t_X and that of A in t_A. In the way in which time intervals are extracted for an item set it is clear that in the time periods from $[t_1', t_1]$ and $[t_2, t_2']$, X does not occur frequently i.e. X occurs rarely. So in calculating a confidence of the rule $A \Rightarrow X\text{-}A$ in $[t_1', t_2']$ if we consider the ratio $Supp_{[t_1,t_2]}X \Big/ Supp_{[t_1',t_2']}A$ instead of

$Supp_{[t_1',t_2']}X \Big/ Supp_{[t_1',t_2']}A$ and lower down the minimum confidence value a bit we hope not to miss any of the rules. This will hold for any set-subset pair, i.e. for each frequent time period of a set there will be a corresponding frequent time period for the subset which includes the former time period.

The above procedure has to be carried out for all locally frequent item set for all its frequent time periods to compute association rules that hold locally. Such a rule $A \Rightarrow X\text{-}A$ can be interpreted as that within a dense region of A in $[t_1', t_2']$ there is a dense region of X in $[t_1, t_2]$ where $[t_1, t_2]$ is a subinterval of $[t_1', t_2']$. We call such rules as local association rule holding in a time interval. We give the algorithm below for finding local association rules.

4. ALGORITHM FOR FINDING LOCAL ASSOCIATION RULES

ALGORITHM

```
S is a set and s is a subset of S
listS ← list of time intervals maintained with S
lists ← list of time intervals maintained with s
while ((pS = listS.get())!= null)
   {tS = pS.ti();
    suppS =  support of the interval tS
    while((ps = lists.get()) != null)
          {ts =  ps.ti();
           if (ts ⊇ tS) break
          }
   supps ← support of s in the interval ts
   if (suppS / supps ≥ minconf) then output
          s ⇒ S – s is an association rule holding in
ts
  }
/* this procedure will require one pass through each
of the lists listS and lists */
```

The function ti() returns the time interval associated with the corresponding node in the time interval lists. The function get () is the member function of class lists described in section-3.4.1. The algorithm is repeated to find the local association rules of the type $s \Rightarrow S - s$ for every possible subsets s of a locally frequent item set S starting from largest possible subset of S. Suppose that the size of S is n then first of all, the algorithm is applied to find the local association rules from all possible (n-1)-size subsets of S to all possible singletons set of S and then from all possible (n-2)-size subsets of S to all possible subset of S of size-2 and so on. If in a particular level a rule from a particular subset of S is not confident then the rules from all the subsets of that particular subset of S will not be confident. This way the procedure is optimized. And the above procedure is repeated for every locally frequent item set obtained using the algorithm.

5. EXPRIMENTAL ESTIMATES FOR THE AMOUNT OF WORK DONE FOR THE ABOVE ALGORITHMS

The problem of mining association rule for market basket problem is known to be NP-complete [15]. In [40], some results about the computational complexity of mining frequent item sets under combined constraints on the number of items and on the frequency threshold is given. In association rule mining the major step is to find the frequent sets. Several implementations of mining frequent item sets are available [7], [16] and [17]. For implementation of our proposed algorithm for mining locally frequent item sets, we have written our codes using a trie based approach. We do not claim our code to be very efficient because there are scopes for improvement. Our aim is to show that the proposed algorithm extracts more rules than those extracted by other known methods. In addition to the usual (non-temporal) frequent item set mining process, the propose algorithm does some more work in keeping the track of the timestamps and in maintaining and manipulating the lists of time intervals associated with the item sets. For counting supports for locally frequent item sets the algorithm will have to do just a few additional operations. For each item sets the value of the last seen can be accessed in O(1) time. The algorithm will have to find the time gap between last seen and the current date and then either start a new time interval or increase the support value for the current time interval. This process will only take up a constant amount of time since for the lists always a pointer to the last node is kept.

Table 1: A partial view of generating association rules from retail dataset

100%-representative Association Rules	Corresponding intervals where the rules hold
39⇒32	[2-1-2000, 22-3-2003]
{38, 39}⇒{41}	[2-1-2000, 25-5-2001]
{38, 41}⇒{39}	[2-1-2000, 25-5-2001], [31-8-2002, 22-3-2003]
{41, 48}⇒{39}	[2-1-2000, 29-5-2001], [31-8-2002, 22-3-2003]
75%-representative Association Rules	**Corresponding intervals where the rules hold**
{39}⇒{41}	[2-1-2000, 29-5-2001], [31-8-2002, 22-3-2003]
{12935}⇒{39}	[13-2-2002, 22-3-2003]
{48}⇒{41}	[31-8-2002, 22-3-2003]
{12935}⇒{48}	[13-2-2002, 22-3-2003]
{41, 48}⇒{39}	[2-1-2000, 29-5-2001]
50%-representative Association Rules	**Corresponding intervals where the rules hold**
{32}⇒{39}	[2-1-2000, 22-3-2003]
{32}⇒{48}	[21-1-2000, 22-3-2003]
{41}⇒{48}	[31-8-2002, 22-3-2003]
{32, 39}⇒{48}	[2-1-2000, 22-3-2003]
{32, 48}⇒{39}	[2-1-2000, 22-3-2003]
{38}⇒{39, 41}	[2-1-2000, 25-5-2001], [31-8-2002, 22-3-2003]
{41}⇒{39, 48}	[2-1-2000, 29-5-2001], [31-8-2002, 22-3-2003]
{39, 41}⇒{48}	[2-1-2000, 29-5-2001], [31-8-2002, 22-3-2003]
{41, 48}⇒{39}	[2-1-2000, 29-5-2001], [31-8-2002, 22-3-2003]
25%-representative Association Rules	**Corresponding intervals where the rules hold**
{41}⇒{32}	[2-1-2000, 25-5-2003]
{41}⇒{32}	[31-8-2002, 22-3-2003]
{48}⇒{12935}	[10-2-2002, 24-4-2002]
{32}⇒{39, 48}	[2-1-2000, 22-3-2003]
{39, 41] ⇒{38}	[2-1-2000, 25-5-2001], [31-8-2002, 22-3-2003]

In candidate generation in addition to the usual process this algorithm will have to compute pair-wire intersection of the two time-interval lists associated with the two item sets that are taking part in the joint operation. This will take O(l+l') time where l and l' are the lengths of the time-intervals lists. But this process will also prune candidates while these are being generated. Similarly, in the pruning step also pairwise intersection of the time-interval lists are to be carried out which requires one pass through the time-interval lists maintained with each subset of the item set under consideration. But this process will also prune more sets than the usual process. In finding local association rules, the proposed method makes a pass through the time-interval lists associated with the set-subset pair of frequent item sets and computes the ratio of the support values. A partial view of the generated association rules from retail dataset is shown in table 1.

6. CONCLUION AND FUTURE WORK

In this paper, we have presented a novel theory for finding local association rules from locally most occurring item sets using rough set and Boolean reasoning. As the discussed algorithm [8], automatically gives all locally most occurring item sets along with a list intervals, there is a hindrance in finding the confidence of an association rules. This paper basically explains the problem in details and presents another option using rough set theory and Boolean reasoning. To find an association rule from a locally most occurring itemset in the interval [t, t′], we chunk down the data, i.e., we consider all those transactions which take place between t and t′ which forms the information system. Thereafter, the information system is converted into a decision table to calculate the reducts. The exactness about the method is that it can accomplish all c-representative association rules. Today, we are working spatial and spatio-temporal data to extract local association rules.

7. REFERENCES

[1] Agrawal, R., Imielinski, T., and Swami, A. N.; Mining association rules between sets of items in large databases, In Proc. of 1993 ACM SIGMOD Int'l Conf on Management of Data, Vol. 22(2) of SIGMOD Records, ACM Press, (1993), pp 207-216.

[2] Agrawal, R., and Srikant, R.; Fast Algorithms for Mining Association Rules, In Proc. of the 20th VLDB Conf., Santiago, Chile, 1994.

[3] Ale, J. M., and Rossi, G. H.; An Approach to Discovering Temporal Association Rules, In Proc. of 2000 ACM symposium on Applied Computing (2000).

[4] Antunes, C. M., and Oliviera, A. L.; Temporal Data Mining an overview, Workshop on Temporal Data Mining-7th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining, (2001).

[5] Chen, X. and Petrounias, I.; A framework for Temporal Data Mining; Proceedins of the 9th International Conference on Databases and Expert Systems Applications, DEXA '98, Vienna, Austria. Springer-Verlag, Berlin; Lecture Notes in Computer Science 1460 (1998) 796-805.

[6] Chen, X. and Petrounias, I.; Language support for Temporal Data Mining; In Proceedings of 2nd European Symposium on Principles of Data Mining and Knowledge Discovery, PKDD '98, Springer Verlag, Berlin (1998) 282-290.

[7] Chen, X., Petrounias, I., and Healthfield, H.; Discovering temporal Association rules in temporal databases; In Proceedings of IADT'98 (International Workshop on Issues and Applications of Database Technology (1998) 312-319.

[8] Mahanta, A. K., Mazarbhuiya, F. A., and Baruah, H. K.; Finding Locally and Periodically Frequent Sets and Periodic Association Rules, In Proc. of 1st Int'l Conf. on Pattern Recognition and Machine Intelligence, LNCS 3776 (2005), pp. 576-582.

[9] Manilla, H., Toivonen, H. and Verkamo, I.; Discovery of frequent episodes in event sequences, Data Mining and

Knowledge Discovery: An International Journal 1(3), (1997). pp. 259-289.

[10] Motwani R., Cohen, E., Datar, M., Fujiware, S., Gionis, A., Indyk, P., Ullman, J. D., and Yang, C.; Finding interesting association rules without support pruning, In Proceedings of the16th International Conference on Data Engineering (ICDE), IEEE (2000).

[11] Ramaswamy, S., Mahajan, S. and Silberschatz, A.; On the discovery of interesting patterns in association rules, In Proc. of 1998 Int'l Conf. on Very Large Databases, (1998), pp. 368-379.

[12] Roddick, J. F., and Spillopoulou, M.; A Biblography of Temporal, Spatial and Spatio-Temporal Data Mining Research, ACM SIGKDD, (June'1999).

[13] Paulak, Z. "Rough Sets In Theoretical Aspects of Reasoning about Data", Kluwer, Netherland, 1991.

[14] Nguyen, H.S. and Nguyen S.H.; Rough Sets and Association Rule Generation, Fudamenta Informaticae 34 (1999), 1-23.

[15] Mazarbhuiya F.A., Mahanta A.K. and Baruah H. K.; Mining Temporal Patterns in Datasets, Ph. D. thesis, Gauhati University, India, 2007.

[16] Nguyen H.S., Slezak D.; Approximate Reducts and Association Rules- Correspondence and Complexity Results. Proc. Of 7th Int'l Workshops on Rough sets, Fuzzy sets and Granular Soft Computing (RSFDGrC'96), Yamaguchi, Janan, 1996.

[17] Skowron A., Rauszer C.; The dicernibility matrices and functions in information systems, In: R. Slowinski (ed.): Intelligent Decision support, Handbook of Applications and Advances of the Rough Sets Theory, Kluwer, Dordrecht, pp. 331-362, 1992.

[18] Wrbewski J.; Covering with reducts- a fast algorithm for rule generation; Proc. of RSCTC'98, Warsaw, Poland, 1998.