Research Journal of Recent Sciences _____ ISSN 2277-2502
Vol. **1(2),** 108-112, Feb. **(2012)** *Res.J.Recent Sci.*

Review Paper

Overview of Non-redundant Association Rule Mining

Shrivastava Neeraj[1] and Lodhi Singh Swati[2]
IES, IPS Academy Indore, MP, INDIA

Available online at: www.isca.in
(Received 5th January 2012, revised 16th January 2012, accepted 21st January 2012)

Abstract

Sequential association rule mining is one of the possible methods to analysis of data. As conventional sequential association rule mining very often generates a huge number of association rules, of which many are redundant, it is desirable to find a solution to get rid of those unnecessary association rules, because of the complexity and temporal ordered characteristics of sequential data, current research of sequential rule mining is limited. Although several sequential association rule prediction model using either sequence constraint or temporal constraint have been proposed, none of them considered the redundancy problem in rule mining. The main purpose of this paper to propose a non redundant sequential association rule mining method proposed the Sequential Min-Max basis for concise representation of non-redundant sequential association rules.

Keywords: Association rule mining, sequential min-max, closed sequence, sequence generator, non-redundant sequential rule mining.

Introduction

Association rule discovery, a successful and important mining task, aims at uncovering all frequent patterns among transactions composed of data attributes or items. Results are presented in the form of rules between different sets of items, along with metrics like the joint and conditional probabilities of the antecedent and consequent, to judge a rule's importance. A closed set contains its own boundary. In other words, if you are "outside" a closed set, you may move a small amount in any direction and still stay outside the set. Note that this is also true if the boundary is the empty set, e.g. in the metric space of rational numbers, for the set of numbers of which the square is less than two. Any intersection of closed sets is closed (including intersections of infinitely many closed sets), and any union of finitely many closed sets is closed. In particular, the empty set and the whole space are closed. In fact, given a set X and a collection F of subsets of X that has these properties, and then F will be the collection of closed sets for a unique topology on X. The intersection property also allows one to define the closure of a set A in a space X, which is defined as the smallest closed subset of X that is a superset of A. Specifically, the closure of A can be constructed as the intersection of all of these closed supersets. Sequential pattern mining, which is the process of extracting certain sequential patterns whose support exceeds a predefined minimal support threshold, has been studied widely in the last decade in the data mining community However, and less work, has been done on sequential association rule mining[1].

Only in recent years, several prediction models which introduced the concept of sequential association rule mining have been proposed, most of which use sequence and temporal constraints in generating association rules. In classical association rule mining, the resulting rule set can easily contain thousands of rules of which many are redundant and thus useless in practice. While in the case of sequential association rule mining, things get even worse This is because the same set of items with different ordering yields different sequential patterns in sequential pattern mining which makes the number of frequent sequential patterns usually much larger than the number of frequent item sets generated from a dataset of a similar size. It is widely recognized that the set of association rules can rapidly grow to be unwieldy, especially as we lower the frequency requirements. The larger the set of frequent item sets the more the number of rules presented to the user, many of which are redundant. This is true even for sparse datasets, but for dense datasets it is simply not feasible to mine all possible frequent item sets, let alone to generate rules between item sets. In such datasets one typically finds an exponential number of frequent item sets.

In many application such as web log data, system traces, purchase histories, financial market data typically is represented in the form of sequences. Sequence data can be a consequence of employing a natural temporal ordering among. An important research in the field of data mining, recently mining sequential data has drawn more and more attention to researchers in the data mining field. In the last decade, many algorithms and techniques have been proposed to deal with the problem of sequential pattern mining including. A priority-based approaches such as GSP and SPADE and pattern-growth based approaches such as Prefix Span[2] and Spam. These existing approaches mainly discuss

how to efficiently generate sequential patterns, and do not pay much attention to the quality of the discovered patterns, in particular, all of these approaches suffer from the problem that the volume of the discovered patterns and association rules could be exceedingly large, but many of the patterns and rules are actually redundant and thus need to be pruned.

Related Work: Recently, some researchers have proposed definition of association rule mining it is a way to find interesting associations among large sets of data items. Association rule mining, which aims to extract interesting correlations and associations among sets of items in large datasets, has two phases: extracting frequent item sets and generating association rules from the frequent item sets with the constraints of minimal support and minimal confidence. The rules with a confidence value larger than a user-specified minimum confidence threshold are considered interesting or useful. There are two basic measures used in association rule mining, support and confidence. Support is a measure that defines the percentage/fraction of records/entries in the dataset that contain $X \cup Y$ to the total number of records/entries. Confidence is a measure that defines the percentage/fraction of records/entries in the dataset that contain $X \cup Y$ to the total number of records/entries that contain just X. The confidence value serves as a measure of the strength or precision of the rule. Apriori is the first association rule mining algorithm that pioneered the use of support-based pruning to systematically control the exponential growth of candidate item sets .In computer science and data mining, Apriori[3] is a classic algorithm for learning association rules. Apriori is designed to operate on databases containing transactions (for example, collections of items bought by customers, or details of a website frequentation). Other algorithms are designed for finding association rules in data having no transactions, or having no timestamps (DNA sequencing). As is common in association rule mining, given a set of item sets (for instance, sets of retail transactions, each listing individual items purchased), the algorithm attempts to find subsets which are common to at least a minimum number C of the item sets. Apriori uses a "bottom up" approach, where frequent subsets are extended one item at a time (a step known as candidate generation) and groups of candidates are tested against the data. The algorithm terminates when no further successful extensions are found. Apriori uses breadth-first search search and a tree structure to count candidate item sets efficiently. It generates candidate item sets of length k from item sets of length $k - 1$. Then it prunes the candidates which have an infrequent sub pattern. According to the downward closure lemma, the candidate set contains all frequent k-length item sets. After that, it scans the transaction database to determine frequent item sets among the candidates.

Example: A large supermarket tracks sales data by stock-keeping unit (SKU) for each item, and thus is able to know what items are typically purchased together. Apriori is a moderately efficient way to build a list of frequent purchased item pairs from this data. Let the database of transactions consist of the sets {1,2,3,4}, {1,2}, {2,3,4}, {2,3}, {1,2,4}, {3,4}, and {2,4}. Each number corresponds to a product such as "butter" or "bread". The first step of Apriori is to count up the frequencies, called the supports, of each member item separately: This table explains the working of Apriori algorithm.

Item	Support
1	3
2	6
3	4
4	5

We can define a minimum support level to qualify as "frequent," which depends on the context. For this case, let min support = 3. Therefore, all are frequent. The next step is to generate a list of all 2-pairs of the frequent items. Had any of the above items not been frequent, they wouldn't have been included as a possible member of possible 2-item pairs. In this way, Apriori prunes the tree of all possible sets. In next step we again select only these items (now 2-pairs are items) which are frequent:

Item	Support
{1,2}	3
{2,3}	3
{2,4}	4
{3,4}	3

And generate a list of all 3-triples of the frequent items (by connecting frequent pair with frequent single item). In the example, there are no frequent 3-triples. Most common 3-triples are {1, 2, 4} and {2, 3, 4}, but their support is equal to 2 which is smaller than our min support. Since all of the Apriori-based mining algorithms have time or space costing problems when handling a huge number of candidate sets and a large database, a new method which avoids candidate generation-and-test and utilizes a new data structure to reduce cost . It is FP-Tree algorithm. The basic idea of the FP-growth algorithm can be described as a recursive elimination scheme: in a preprocessing step delete all items from the transactions that are not frequent individually, i.e., do not appear in a user-specified minimum number of transactions. Pattern-growth approach is more efficient and scalable than other approaches, such as Apriori[3], and is effective in mining dense database. Sequential pattern mining[5] is the process of extracting certain sequential patterns whose support exceeds a predefined minimal support threshold. Since the number of sequences can be very large, and users have different interests and requirements, to get the most interesting sequential patterns usually a minimum support is predefined by the users. By

using the minimum support we can prune out those sequential patterns of no interest, consequently making the mining process more efficient. Introduced another metric called surprise to measure the interestingness of sequences.

Methodology

In this paper we give the algorithm of non redundant association rule mining. From sequential patterns, we can extract relations between the sets of sequences in the format of sequential association rules. However, the huge number of sequential rules will cause several problems. The first issue is the large quantity of low quality rules that are almost meaningless and will not give people any useful information. The second issue is that the rule generation cost of the full sequential rules is also quite high, even for a sparse dataset. Moreover, sometimes it is even impossible to use for mining full sequential rules for dense datasets. So it is quite reasonable to seek a concise and non-redundant representation for sequential rules.

Non redundant association rule mining: A non-redundant sequential mining method is adapted to extract valuable information from the user navigation sessions generated by web usage mining technique (we omit the details due to the page limit). In non-sequential association rule miningfield[6] defined a condensed representation for association rules; the representation was characterized by frequent closed item sets and their generators. And the research aimed to present rules with minimal antecedent and maximal consequent. This technique can remove significant amount of redundant association rules to help improve the quality of the mining result. We extend the non-redundant association rule theory into the sequential mining area. We first give the definitions of the redundant sequential rules, and then we introduce the min-max sequential basis for non-redundant rule representation which requires both sequential generators and closed sequential patterns. Finally we provide an algorithm which can efficiently extract non-redundant sequential rules.

Definition: (Redundant Sequential Association Rules): Let X → Y and X' →Y' be two sequential rules with confidence cf and cf', respectively. X →Y is said a redundant rule to X' →Y' if X' belong to X ; Y belong to Y', and cf ≤ cf'.

Definition: Confidence of sequential rules: Let X, Y is two different sequences in a transaction database. The confidence of rule X U Y is defined as support (X, Y)/support(X). Note that support(X, Y) represents the number of sequences that contain both X and Y in transaction database. Sequential Min-max rules Sequential rules share the same characteristics or key concepts as non-sequential rules, such as transitivity, support, and confidence. The only difference is that sequential associations describe sequential relationships between items with the redundant rule definition we extend the min-max non-redundant sequential rule theory from association rule mining to the Sequential association rule mining area.

Definition (Min-max sequential association rules): Let *R* be the set of sequential association rules. A sequential rule r: $s1 \rightarrow s2 \in R$ is a min-max sequential rule if not $\exists \rho \rightarrow: s1' \rightarrow s2' \in R$ with support(s') = support(s); confidence(s') = confidence(s), $s \sqsubseteq s1$, and $s2' \sqsubseteq s2.1'$.The min-max sequential association rules are the non-redundant sequential rules having minimal antecedent and maximal consequent. r is a min-max sequential rule if no other sequential rule r' has the same support and confidence, and it has an antecedent that is the subsequence of the antecedent of s and a consequent that is a super sequence of the consequent of r. We define two new bases which suit sequential data. They are sequential min-max exact basis and sequential min-max approximate basis. These two bases also formed a concise representation for sequential association rules. Suppose that g, s1, s2 are sequences, g ⊏ s1 ⊏ s2, closure (g) = closure (s1) = closures (s2), then the two rules r1: g→ (s2\g) and r2: s1→ (s2\s1), where s2\s1 denotes a subsequence of s2 by removing the sequence s1, will have the same confidence, the antecedent of r1 is shorter than that of r2 and the consequent of r1 is longer than that of r2. If .s2 is a closed sequence and g is a generator, i.e., s2=closure (s2) and g is the minimal sequence which has the same closure as s2, g⊏ (s2\g) will have the shortest antecedent and longest consequent among the rules s1⊏ (s2\s1) where g ⊑s1 ⊏ s2. Therefore, similar to Min-max exact rules which are generated using a closed item set and its generator, the sequential exact rules can be generated using a closed sequence and its sequential generators Since closure(g) =closure(s2), the confidence of g⊏ (s2\g) is 1. The sequential Min-max exact rules are defined as follows.

Definition (Sequential Min-Max Exact Basis): Let Closed be the set of closed sequential patterns, and for each closed sequential pattern c, let Genc be the set of sequential generators of c. The sequential min-max exact basis is:
Sequential Min Max Exact = $\{r: g \rightarrow (c\backslash g) \mid c \in Closed \wedge g \in Genc \wedge g \neq c\}$

Definition (Sequential Min-Max Approximate Basis): Let Closed be the set of closed sequential patterns, Gen be the set of all sequential generators of closed sequential patterns in Closed. The sequential min-max approximate basis is:
Sequential Min Max Approx = $\{r: g \rightarrow (c\backslash g) \mid c \in Closed \wedge g \in Gen \wedge$ Closure (g) $\neq c\}$

Rule generation based on closed sequences and sequence generators: The non-redundant sequential rules are generated from the frequent closed sequences and the sequential generators According to the definition of sequential min-max rules; we use the sequential generators as

antecedents of rules. For each generator, we generate its consequent in the form of removing the same prefix parts from a closed sequence which the generator has the remaining part of this closed sequence is considered as the correspondent consequent of the rule.

Algorithm: NR Rule Mining (L, min_sup, min_confidence)
Input: min_sup, min-confidence, Closed sequence set L including corresponding generators
Output: Rule set R (non-redundant sequential rules)
i. Let C = closed sequence set of L, G = generator set of L
ii. For each sequence pattern $g \in \Gamma$
iii. For each closed sequence pattern $c \in \Lambda$
iv. b: = c/g
v. r: $g \rightarrow \beta$, support(r) = support (c), and confidence(r) = support(c)/support (g)
6 If (confidence(r) >= min_confidence)
7 Add rule r into rule set R
For example, if sequence AB is in a generator, while sequence ABDF is a closed sequence, then AB U DF is considered to be a non-redundant sequential rule.

Simulation and Results

For an experimental evaluation of the proposed algorithms, we performed several experiments on real datasets we implemented the proposed algorithms in C++. All datasets were obtained from the UCI Machine Learning Repository. The chess and bread datasets are derived from their respective market analysis; the butter database contains characteristics of various test of butter. In particular, a large amount of rules can be derived exactly. Some of the results are also given in numerical form in table 1. The table reports results for exactly derivable rules with identical consequent sub rules, with identical condition or consequent subrules, or with allsubrules. The row "1% interval" was obtained by pruning rules for which the lower and upper bounds of confidence are at most 1 percentage point apart. Results with minimal closed rules are included for comparison. The number of non-derivable association rules is less than the number of minimal closed rules already when using only subrules with identical consequent or condition in chess and connect datasets. In butter the number of non-derivable association rules is less than the number of minimal closed rules if we use all subrules to compute the upper and lower bound. In chess the number of minimal closed rules is slightly less than the number of non-derivable association rules. Relatively small error bounds, already in the order of fractions of percent, can result in significant further pruning. For example in the chess dataset, the number of nonderivable association rules when using all subrules becomes less than the number of minimal closed rules when we allow the difference of upper and lower bound to be one percentage unit. In other datasets the effect of allowing a small interval for the confidence bounds is even more radical. Reducing the extraction to non-transitive rules in the proper basis for approximate rules can also be interesting. Datasets the average relative size of bases compared with the sets of all rules obtained. In the case of weakly correlated data, no exact rule is generated and the proper basis for approximate rules contains all approximate rules that hold. The reason is that, in such data, all frequent item sets are frequent closed item sets. In the case of correlated data, the number of extracted rules in bases is much smaller than the total number of rules that hold. Dataset the execution times of the computation of all rules. Execution times of the derivation of the exact rules and the proper basis for non-transitive approximate rules are not presented since they are identical, as shown in table 1.

Table-1
Number of rules after different pruning methods

	Bread	**Chess**	**Butter**
All rules	8160110 100%	3667800 100%	1429200 100%
Identical consequent	1550420 19%	557579 15%	695871 49%
Id. Condition or consequent	57936 .71%	11200 .28%	177155 12%
All subrules	3345 .041%	552 .014%	543 .038%
All subrules 1% interval	63 .0078%	160 .0043%	16345 1.1%
Minimal closed rules	138721 1.7%	15496 .42%	71813 5%

Advantages: (i) Mining in transitive reduction to avoid substantial space and I/O overhead. (ii) Directly extracting frequent closed partial orders in transitive reduction. (iii) The user is provided with a smaller set of resulting rules, easier to handle, and information of improved quality. (iv) Execution times are reduced compared with the discovering of all association rules. (v) The number of rules is minimal; moreover these rules have minimal antecedent and maximal consequent.

Conclusion

We presented the definition of the redundancy of sequential association rule and proposed the Sequential Min-Max basis for concise representation of non-redundant sequential association rules. According to this basis, we introduced a method for mining non-redundant sequential rules based on sequential generators and closed sequential patterns. Our method guaranteed the generated non-redundant sequential rules have the minimal antecedent and the maximal consequent. Nevertheless, in future work, we will explore a sequential pruning mechanism in which only subrules are used that is confident and that where not already pruned earlier.

References

1. Ayres J., Flannick J., Gehrke J. and Yiu T., Sequential PAttern Mining using a Bitmap Representation. Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining **(2002)**

2. Agrawal R. and Srikant R., Fast algorithms for mining association rules in large databases, Proceedings of 20th International Conference on Very Large Databases **(1994)**

3. Desikan P., Pathak N., Srivastava J. and Kumar V., Incremental page rank computation on evolving graphs. Paper presented at the Special interest tracks and posters of the 14th International Conference on World Wide Web **(2005)**

4. Ganter B. and Wille R., Formal Concept Analysis: Mathematical Foundations, Springer, Berlin-Heidelberg-New York, **10, (1999)**

5. Gaul W. and Schmidt-Thieme L., Mining Generalized Association Rules for Sequential and Path Data, Proceedings of the 2001 IEEE International Conference on Data Mining **(2001)**

6. Agrawal R., Imielinski T. and Swami A., Mining association rules between sets of items in large databases, Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data **(1993)**

7. Agrawal R. and Srikant R., Mining sequential patterns, Proceedings of the Eleventh International Conference on Data Engineering 1995 **(1995)**

8. Ashrafi M.Z., Taniar D. and Smith K., Redundant association rules reduction techniques, International Journal of Business Intelligence and Data Mining **(2007)**

9. Guo S., Liang Y., Zhang Z. and Liu W., Association Rule Retrieved from Web Log Based on Rough Set Theory. Proceedings of the Fourth International Conference on Fuzzy Systems and Knowledge Discovery, **03, (2007)**

10. Brin S., Motwani R., Ullman J.D., and Tsur S., Dynamic item set counting and implication rules for market basket data, In SIGMOD 1997, Proceedings ACM SIGMOD International Conference on Management of Data, May 13-15, 1997, 255-264 (**1997)**