International Journal of Information Technology and Knowledge Management
July-December 2008, Volume 1, No. 2, pp. 191-198

MINING FUZZY GENERALIZED ASSOCIATION RULES FOR ER MODELS

Praveen Arora, Ram Kumar & Ashwani Kush

Abstract

Data mining techniques can be implemented rapidly on existing software and hardware platforms. Some algorithms like Extended Apriori and Apriori star exist to discover the relationships between data attributes upon all levels of fuzzy taxonomic structures that exist in single table. The purpose of the paper is to address the issue of mining fuzzy association rules in databases designed using Entity-Relationship (ER) Models at multiple levels. The study aims to extend the previous developed algorithms Extended Apriori and Apriori star to discover a new algorithm. The study will help in standardizing algorithms for finding appropriate results from database tables containing fuzzy data.

1. Introduction

Data mining is extracting or mining knowledge from large amounts of data. That is, it is the process of extracting useful and interesting patterns from large datasets to increase the potential of knowledge. Data mining techniques can be implemented rapidly on existing software and hardware platforms to enhance the value of existing information resources, and can be integrated with new products and systems as they are brought on-line. Non-trivial extraction of novel, implicit and actionable knowledge from extremely large data sets **[12]**. One popular summarization and pattern extraction algorithm is the association rule algorithm Apriori **[3]**, which identifies correlations between items in transactional databases. Association Rules mining finds interesting association or correlation relationships among a large set of data items. With massive amounts of data continuously being collected and stored, many industries are becoming interested in mining association rules from their databases. The discovery of interested association relationships among huge amounts of business transaction records can help in many business decision-making processes **[11]**. The problem of mining association rules is to find all rules that satisfy a user-specified minimum support and minimum confidence.

Most of the current data mining algorithms handle databases consisting of single table. Though, data mining is generally performed on data stored in data warehouses, even so the data may not be stored in a single table. For instance, the star database design used in data warehouses is derived from entity-relationship design. The data is broadly organized into two groups-entities and relationships. Each entity table stores information on all attributes associated with the particular entity and relationship table

represents relationship among different entities. If traditional data mining algorithms are used to discover association rules in such environments then first a join of entity tables and relationship table needs to be computed which in turn adversely affects the efficiency and cost of the algorithm used. It has been observed that much of the research has been done for mining fuzzy generalized association rules, mining association rules in Entity–Relationship modeled Databases. Less work has been done on development of fuzzy data mining association rules for multiple tables.

2. RECENT STUDIES

The problem of mining association rules has been discussed by Agrawal et al **[1]**. This study presents an efficient algorithm that generates all significant association rules between sets of items in a large database of customer transactions. It is an AIS algorithm known to be the first published algorithm to generate all large itemsets in a transaction database. It focuses on the enhancement of databases with necessary functionality to process decision support queries. This algorithm has been targeted to discover qualitative rules. This technique is limited to only one item in the consequence.

The AIS algorithm makes multiple passes over the entire database. During each pass, it scans all transactions. In the first pass, it counts the support of individual items and determines which of them are large or frequent in the database. Large itemsets of each pass are extended to generate candidate itemsets. After scanning a transaction, the common itemsets between large itemsets of the previous pass and items of this transaction are determined. These common itemsets are extended with other items in the transaction to generate new candidate itemsets. A large itemset *l* is extended with only those items in the transaction that are large and occur in the lexicographic ordering of items later than any of the items in *l*. To perform this task efficiently, it uses an estimation tool and pruning technique.

The disadvantage is that this results in unnecessarily generating and counting too many *candidate* itemsets that turn out to be *small*.

Set–Oriented Mining for association rules in relational Databases is described by Swami **[2]** where an algorithm **SETM** has been developed with the desire to use SQL to compute large itemsets. In this algorithm each member of the set large itemsets, $\overline{L_k}$, is in the form <TID, itemset> where TID is the unique identifier of a transaction. Similarly, each member of the set of candidate itemsets, $\overline{C_k}$, is in the form <TID, itemset>.

Similar to the AIS algorithm, the SETM algorithm makes multiple passes over the database. In the first pass, it counts the support of individual items and determines which of them are large or frequent in the database. Then, it generates the candidate

itemsets by extending large itemsets of the previous pass. In addition, the SETM remembers the TIDs of the generating transactions with the candidate itemsets. The relational merge-join operation can be used to generate candidate itemsets. Generating candidate sets, the SETM algorithm saves a copy of the candidate itemsets together with TID of the generating transaction in a sequential manner. Afterwards, the candidate itemsets are sorted on itemsets, and small itemsets are deleted by using an aggregation function. If the database is in sorted order on the basis of TID, large itemsets contained in a transaction in the next pass are obtained by sorting $\overline{L_k}$ on TID. This way, several passes are made on the database. When no more large itemsets are found, the algorithm terminates.

In addition to having same disadvantage as of the AIS algorithm, also it is that for each *candidate* itemset, there are as many entries as its support value.

Apriori and *AprioriTid* algorithms **[3]** are used to discover association rules between items in a large database of sales transactions. Results show that these algorithms always outperform the earlier algorithms **AIS** and **SETM**. The study also show how the best features of the two proposed algorithms can be combined into a hybrid algorithm, called AprioriHybrid. Experiments reveal that AprioriHybrid scales linearly with the number of transactions. The execution times decrease little as the number of items in the database increases. As the average transaction size increases, the execution times increase only gradually. In another study by Mannila **[4]** the properties of association rule discovery in relations has been discussed. An algorithm has been proposed for the problem that outperforms the method in [1] by a factor of 5. The algorithm has been based on the same basic idea of repeated passes over the database as the method in [1] with the difference that this algorithm makes careful use of the combinatorial information obtained from previous passes and in this way avoids considering many unnecessary sets in the process of finding the association rules. Experimental results of the algorithm shows improvement when compared against the previous results, and is also simple to implement. The study also shows that sampling is an efficient technique for finding rules of this type, and that algorithms working in main memory can give extremely good approximations. Studies on mining association rules find rules at single concept level, but mining association rules at multiple concept levels may lead to the discovery of more specific and concrete knowledge from data **[5]**. In this study, a top-down progressive deepening method is developed for mining multiple level association rules from large transaction databases that extends the existing single level association rule mining algorithms and explores techniques for sharing data structures and intermediate results across levels. Concept hierarchy handling, methods for mining flexible multiple-level association rules, and adaptation to difference mining requests are also discussed in the study. Srikant **[6, 7]** introduces the problem of mining generalized association rules where a database of transactions consists of a set of items, and taxonomy

(is-a-hierarchy) on the items. The paper finds associations between items at any level of the taxonomy. The study replaces each transaction with an "extended transaction" that contains all the items in the original transaction as well as all the ancestors of each item in the original transaction. Any of the earlier algorithms are then run on these transactions to get generalized association rules. But this Basic approach has been found to be slow. It presents two algorithms *Cumulate* and *EstMerge* for finding generalized association rules. Both Cumulate and EstMerge algorithms run 2 to 5 times faster than Basic; EstMerge performs somewhat better than Cumulate, with the performance gap increasing as the size of the database increases.

It also introduces the problem of mining association rules in large relational tables containing both quantitative and categorical attributes. It partitions the quantitative attributes and maps the categorical attributes with the integers. But two problems arise when partitioning of quantitative attributes are done. a) MinSup: If intervals are too small, some rules may not have min Support. b) MinConf: If intervals are too large, some rules may not have min Confidence. To overcome aforementioned problems, the adjacent values/intervals were combined to avoid the threshold support problem, and increasing the number of intervals to avoid the threshold confidence problem.

However, this approach leads to two new problems as Higher Execution Time and Many Rules. High Execution time arises when the number of intervals for an attribute is increased. On the other hand, increased number of rules are obtained, if any range is considered that contains the interval satisfying the threshold support. By introducing a user specified "maximum support" parameter, the extension of adjacent values/intervals is restricted. The adjacent values/intervals are combined until the combined support is less than the maximum support. However, any single interval/value whose support exceeds maximum support is still considered. As a result of this, the "Higher Execution Time" problem is reduced to a certain extent. The problem of finding frequent itemsets from the database with quantitative attributes is solved in three steps. At first, decide whether each attribute is to be partitioned or not. If an attribute is to be partitioned, determine the number of partitions. Then, map the values of the attribute to a set of consecutive integers. Afterwards, find support of each value of all attributes. To avoid "Minsup" problem, adjacent values are combined as long as their support is less than user-specified maximum support. All ranges and values with minimum support form the set of frequent itemsets. Solutions presented in this paper do not work well when applied to interval data where separation between data values has meaning. **Kuok [8]** has proposed a method to handle quantitative attributes for which each attribute is assigned several fuzzy sets. Fuzzy sets handle numerical values better than existing methods because fuzzy sets soften the effect of sharp boundaries. The paper proposes an algorithm for mining fuzzy *AR* of the form: If X is A then Y is B. X & Y are attributes and A & B are fuzzy sets which characterize X and Y respectively. The fuzzy set concept

is better than the partition method because fuzzy sets provide a smooth transition between member and non-member of a set. Because of smooth transition, there are fewer boundary elements being excluded. The paper uses Significance and certainty factor to determine the satisfiability of itemsets and rules.

In many real life applications, the related taxonomic structures may not be necessarily crisp, rather certain fuzzy taxonomic structures reflecting partial belonging of one item to another may pertain **[9]**. For example, Carrot may be regarded as being both Fruit and Vegetable, but to different degrees. An example of a fuzzy taxonomic structure is shown in figure 1. Here, a sub-item belongs to its super-item with a certain degree.

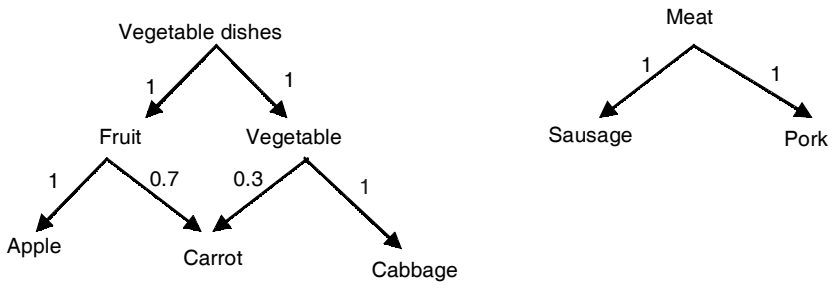

Figure 1: Example of Fuzzy Taxonomic Structures

A crisp taxonomic structure assumes that the child item belongs to its ancestor with degree 1. But in a fuzzy taxonomy; this assumption is no longer true. Different degrees may pertain across all nodes (item sets) of the structure. The study focuses on the issue of mining generalized association rules with fuzzy taxonomic structures. The study extends **Apriori** and **Fast** algorithm to allow discovering the relationships between data attributes upon all levels of fuzzy taxonomic structures. Various sub-algorithms have also been developed. The Extended algorithms were run on 1,000,000 transactions and the algorithms were found to be at he same level of computational complexity as that of the classical algorithm and had the same performance level. Current data mining algorithms **[10]** handle databases consisting of a single table. This study addresses the problem of mining association rules in databases consisting of multiple tables and designed using the entity-relationship model. To address this issue the study introduces the notion of entity and join support and presents two algorithms: algorithm **Apriori Join,** for mining the outer join of a star schema tables using the knowledge of the schema, and algorithm **Apriori Star**, for directly mining the star schema database. A study by Chen **[11]** aims at dealing with the fuzzy association rules of the form $X \rightarrow Y$ where X and Y can be collections of fuzzy sets. It incorporates fuzziness in the exact taxonomies that reflect partial belongings among itemsets. A number of sub-algorithms as Apriori fast algorithms (GAR), an algorithm to deal with fuzzy taxonomies (FGAR),

An algorithm to deal with linguistic hedges (HFGAR) have been introduced to express meaningful knowledge in a more natural and abstract way.

3 PROPOSED STUDY

The proposed study focuses on developing an algorithm that will use fuzzy logic for finding fuzzy association rules from ER models. The study will help the management of the Supermarkets in making their business plans that includes what to put on sale, how to design coupons, how to maximize the profits. Analysis of the transaction data which includes the customer personal data and the goods that customers purchase converted into fuzzy taxonomic structures reflecting partial belonging of item to another will be the approach in order to improve the quality of such decisions. The study will focus on finding rules such as: young $\Rightarrow$ Meat which implies that customer of the age group 20-30 and 30-40 might turn to buy Meat where the age group 30-40 partially belongs to Young with degree $\mu_{young30\text{-}40}$ where both young and Meat belong to two different entities designed using ER models. Another example of such a rule can be Height = Tall $\Rightarrow$ Game = Basket Ball which implies that a person with the height 5' 7" – 5' 9" and 6' 0"+ might opt for Basket Ball game where group 5' 7" – 5' 9" partially belongs to Tall with degree $\mu_{Tall5'\ 7"-5'\ 9"}$ and both Height and Game belong to two different entities. In this example the attributes **Height** of the **candidate** table will be first converted into fuzzy taxonomic structures respectively reflecting partial belonging of one item to another. Figure 2 explains the concept.

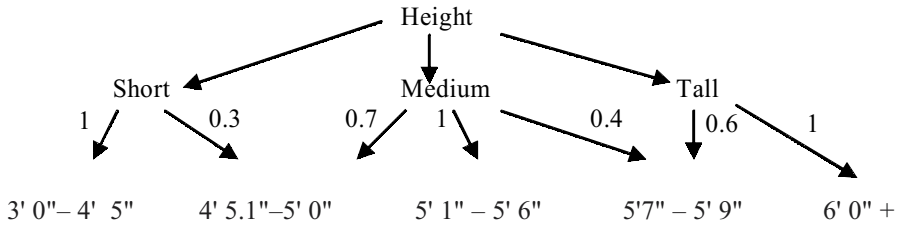

Figure 2: Example of Fuzzy Taxonomic Structure of Height

The paper discovers a new algorithm *Extended Apriori Star* to find such kind of rules. The fuzzy extensions that will be presented in this study will enable us to discover not only crisp generalized association rules but also fuzzy generalized association rules when databases consisting of several tables organized in a schema within the framework of fuzzy taxonomic structures. Strong Association rules between items of fuzzy nature existing in multiple tables can be calculated that will undoubtedly help in understanding things in broad spectrum.

4. CONCLUSION

The Proposed algorithm is discovered by extending the previous algorithms *Extended Apriori* and *Apriori Star* that will be used to find the fuzzy association rules in Entity–Relationship modeled databases, which is capable to handle multiple tables. The study analyzes how the attributes of several entities appear together. The Study also analyzes the rules with respect to the relationships existing between the entities and their ancestors. If several relationships exist between two or more entities, then the fuzzy association rules between their attributes and ancestors are examined with respect to each such relationship. Extended Apriori Star Algorithm, which uses fuzzy logic for finding the fuzzy association rules from multiple tables, will be implemented in VC++. For this implementation the dataset will be taken from supermarkets for the goods that the customers will purchase and customer's personal database will also be taken. The performance of the algorithm will be analyzed comparing it with existing structures.

REFERENCES

[1] R. Agrawal, T. Imielinski, and A. Swami. Mining Association Rules between Sets of Items in Large Databases. *In Proc. of the ACM SIGMOD Conference on Management of Data*, Washington, D.C., May 1993, 207–216.

[2] M. Houtsma and A. Swami. Set-oriented Mining of Association Rules. In *Proc. of the 11th International Conference on Data Engineering*, Taipei, Taiwan, March 1995, 25-33.

[3] R. Agrawal and R. Srikant. Fast Algorithms for Mining Association Rules. In *Proc. of the VLDB Conference*, Santiago, Chile, September 1994. Expanded Version Available as IBM Research Report RJ9839, June 1994.

[4] H. Mannila, H. Toivonen, and A. I. Verkamo. Efficient Algorithms for Discovering Association Rules. In KDD-94: *AAAI Workshop on Knowledge Discovery in Databases*, Seattle, Washington, July 1994, 181–192.

[5] J. Han, Y. Fu, Discovery of Multiple-level Association Rules from Large Databases, *Proceedings of the 21st International Conference on VLDB*, Zurich, Switzerland, September 1995.

[6] R. Srikant and R. Agrawal. Mining Generalized Association Rules, *Proceedings of the 21st VLDB Conference*, Zurich, Switzerland, (1995).

[7] R. Srikant and R. Agarwal, Mining Quantitative Association Rules in Large Relational Tables, In Proceedings of the ACM SIGMOD *International Conference on Management of Data*, Montreal, Quebec, Canada, June 04-06, 1–12.

[8] C H Kuok, A Fu, M H Wong, Mining Fuzzy Association Rules in Databases ACM SIGMOD Record, **27**, (1), ACM Press, (1998).

[9] Chen G., Wei Q. and Kerre E., "*Fuzzy Data Mining: Discovery of Fuzzy Generalized Association Rules*". In Bordagna and Pasi (eds.), Recent Research issues on Management of Fuzziness in Databases, Physica-verlag (Springer), (2000).

[10] Cristofor L. and Simovici D., "Mining Association Rules in Entity-Relationship Modeled Databases", *Technical Report*, UMB, TR 2001-2002.

[11] G. Chen and Q. Wei, Fuzzy Association Rules and the Extending Mining Algorithms, Information Sciences: *An International Journal*, **147**, 201–228.

[12] Han J., Kamber M., "*Data Mining: Concepts and Techniques*", Harcourt India Pvt Ltd., (2001).

[13] W. J. Frawley, G. Piatetsky-Sapiro, and C. J. Matheus, "Knowledge Discovery in Databases: An Overview".

Praveen Arora
JaganNath Institute of Mgmt. Sciences
Delhi
E-mail: praveen@jimsindia.org

Ram Kumar
DCSA
K.U. Kurukshetra
E-mail: rkckuk@rediffmail.com

Ashwani Kush
University College
K.U. Kurukshetra
E-mail: akush20@gmail.com