Association Rule Mining Techniques With Respect to their Privacy Preserving Capabilities

Gopinath Puppala, Research Scholar, Dept. of Computer Science, Maharaja Agrasen Himalayan Garhwal University

Dr. Ajay Kumar Chaurasia, Assistant Professor, Dept. of Computer Science, Maharaja Agrasen Himalayan Garhwal University

ABSTRACT

Data mining, an urging requirement in the current era and whose scope of research is expected to be for upcoming decades. Among the well versed techniques of data mining association rule mining plays a prodigious role. This technique emphasizes on curious association, correlations, frequent patterns etc. from the given data sources to be mined. The primary task of association mining resides in uncovering the frequent patterns and exploring the association rules. Multiple variation of association rule mining algorithms with regard to their performance factors are available. One important constraint entailing the extraction of association rules is, privacy preserving of sensitive data. There is a need to maintain a sustainable ratio between protection of privacy and knowledge discovery. This paper enlightens and reviews the various association mining techniques with respect to their privacy preserving capabilities and also adds a pinch on the tools preferred for such privacy. This paper deals with is a popular and well-accepted method for discovering interesting relations between variables in large databases. An association rule implies certain association relationships among a set of objects Association rules are employed today in many application areas including Web usage mining, intrusion detection and bioinformatics. It analyzes the algorithms of Apriori, Predictive Apriori and Tertius algorithms.

Key words: **Algorithm, bioinformatics, association mining techniques**

INTRODUCTION

In data mining, association rule learning is a popular and well-accepted method for discovering interesting relations between variables in large databases. Association rules are employed today in many areas including web usage mining, intrusion detection and bioinformatics (Dong Liu, et. al., 2015) (Vikas Markam, 2016). Piatetsky-Shapiro describes analyzing and presenting strong rules discovered in databases using different measures of interestingness. Interestingness measures play an important role in data mining, regardless of the kind of patterns being mined. These measures are intended for selecting and ranking patterns according to their potential interest to the user (Lau A, et.al, 2003). Good measures also allow the time and space costs of the mining process to be reduced. Based on the concept of strong rules, association rules for discovering regularities between products in large scale transaction data recorded by point-of-sale (POS) systems in supermarkets was introduced. For example, the rule {onion, vegetables} ={rice} found in the sales data of a supermarket would indicate that if a customer buys onions and vegetables together he is likely to also buy rice. Such information can be used as the basis for decisions about marketing activities such as promotional pricing or product placements.

DATA MINING

The world is deluged with various kinds of data-scientific data, environmental data, financial data and mathematical data. Manually analyzing, classifying, and summarizing the data is impossible because of the incredible increase in data in this age of net work and information sharing. This research investigates the fundamentals of data mining and current research on integrating uncertainty into data mining in an effort to develop new techniques for incorporating uncertainty management in data mining.

Briefly speaking, data mining refers to extracting useful information from vast amounts of data. Many other terms are being used to interpret data mining, such as knowledge mining from databases, knowledge extraction, data analysis, and data archaeology. Nowadays, it is commonly agreed that data mining is an essential step in the process of knowledge discovery in databases, or KDD. In this paper, based on a broad view of data mining functionality, data mining is the process of discovering interesting knowledge from large amounts of data stored either in databases, data warehouses, or other information repositories.

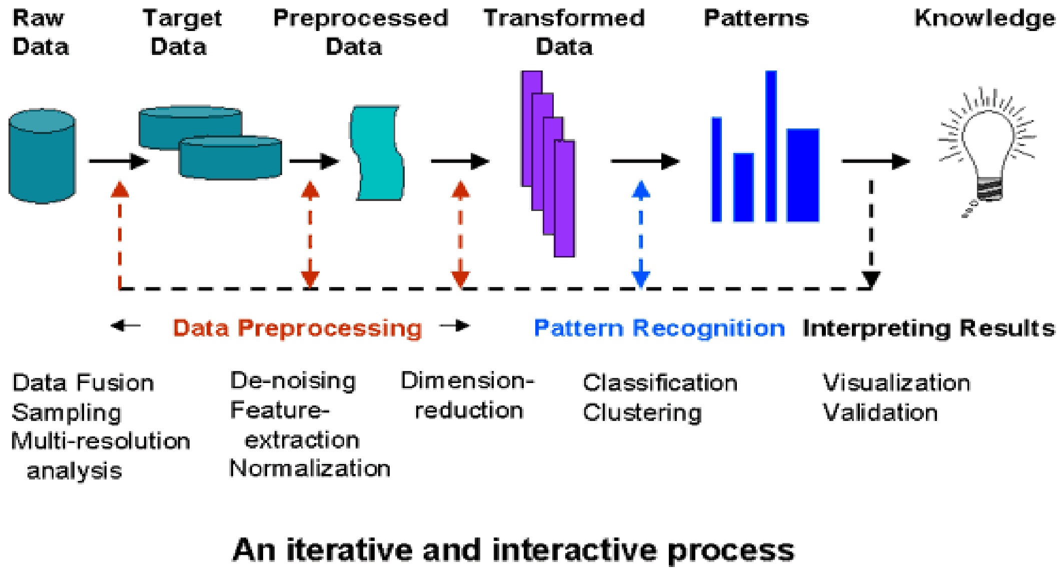


Figure 1.1 Knowledge Discovery in Databases

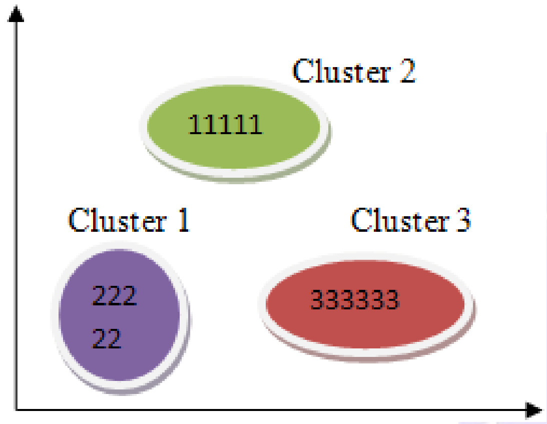
Background

Necessity is the mother of invention. Since ancient times, our ancestors have been searching for useful information from data by hand. However, with the rapidly increasing volume of data in modern times, more automatic and effective mining approaches are required. Early methods such as Bayes' theorem in the 1700s and regression analysis in the 1800s were some of the first techniques used to identify patterns in data. After the 1900s, with the proliferation, ubiquity, and continuously developing power of computer technology, data collection and data storage were remarkably enlarged. As data sets have grown in size and complexity, direct hands-on data analysis has increasingly been augmented with indirect, automatic data processing. This has been aided by other discoveries in computer science, such as neural networks, clustering, genetic algorithms in the 1950s, Decision trees in the 1960s and support vector machines in the 1980s.

Data mining is the process of applying these methods to data with the intention of uncovering hidden patterns. Data mining or data mining technology has been used for many years by many fields such as businesses, scientists and governments. It is used to sift through volumes of data such as airline passenger trip information, population data and marketing data to generate market research reports, although that reporting is sometimes not considered to be data mining.

Data mining commonly involves four classes of tasks: (1) classification, arranges the data into predefined groups; (2) clustering, is like classification but the groups are not predefined, so the algorithm will try to group similar items together; (3) regression, attempting to find a function which models the data with the least error; and (4) association rule learning, searching for relationships between variables.

According to Han and Kamber [2015], data mining functionalities include data characterization, data discrimination, association analysis, classification, clustering, outlier analysis, and data evolution analysis. Data characterization is a summarization of the general characteristics or features of a target class of data. Data discrimination is a comparison of the general features of target class objects with the general features of objects from one or a set of contrasting classes. Association analysis is the discovery of association rules showing attribute-value conditions that occur frequently together in a given set of data. Classification is the process of finding a set of models or functions that describe and distinguish data classes or concepts, for the purpose of being able to use the model to predict the class of objects whose class label is unknown. Clustering analyzes data objects without consulting a known class model. Outlier and data evolution analysis describe and model regularities or trends for objects whose behavior changes over time.



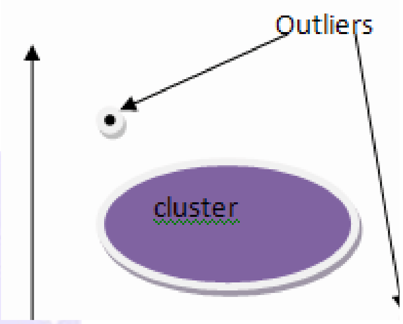


Figure 1.2. Cluster Analysis for numbers **Figure 1.3. Outlier Analysis**

CASE STUDIES

A few case studies pertaining to breast cancer, mushroom, larynx cancer, zoo, sunburn, imaginary disease, contact lenses, MONK, soya bean and titanic datasets are executed to find the utility of association rule mining. The data sets are categorical in nature. The arff conversion of the data set was provided by Håkan Kjellerstrand.

The breast cancer domain was obtained from the University Medical Centre, Institute of Oncology, Ljubljana, Yugoslavia. M. Zwitter and M. Soklic have provided the data (Michalski RS, et.al, 1986). This data set has 286 instances described by 9 attributes + one class attribute. The set includes 201 instances of one class and 85 instances of another class.

The mushroom domain was obtained from The Audubon Society Field Guide to North American Mushrooms. G. H. Lincoff and Alfred A. Knopf have provided the data (Lincoff GH, et.al, 1981). Jeff Schlimmer donated the data to UCI Machine Learning Repository. This data set has 8124 instances described by 22 attributes + one class attribute. This set includes 4208 instances of one class and 3916 instances of another class. The larynx cancer domain was obtained from Mendenhall, WM, Million RR, Sharkey DE and Cassis NJ. (Mendenhall WM, et.al, 1984). This data set has 41 instances described by 1 attribute + one class attribute. The set includes 23 instances of one class and 18 instances of another class.

The zoo domain was obtained from Richard Forsyth. This data set has 101 instances described by 16 attribute + one class attribute. The set includes 58 instances of one class and 43 instances of another class. The sunburn domain was obtained from Patrick Winston (Winston, 1992). This data set has 8 instances described by 4 attribute + one class attribute. The set includes 5 instances of one class and 3 instances of another class.

The imaginary disease domain is derived from Simon Langley (Langley, 1992). This data set has 10 instances described by 4 attribute + one class attribute. The set includes 6 instances of one class and 4 instances of another class.

The contact lens domain was obtained from Cendrowska J and donated by Benoit Julien to UCI Machine Learning Repository. This data set has 24 instances described by 4 attributes + one class attribute. This set includes 4 instances of one class and 5 instances of second class and 15 instances of third class.

The MONK domain was obtained from Sebastian Thrun. This data set has 124 instances described by 6 attribute + one class attribute. The set includes 62 instances of one class and 62 instances of another class.

The soya bean domain was obtained from Michalski RS and Chilausky RL. (Michalski RS, et.al, 1980) and donated by Ming Tang and Jeff Schlimmer to UCI Machine Learning Repository. This data set has 683 instances described by 35 attribute + one class attribute. The set includes 19 classes with the number of instances in the classes being 20, 20, 20, 88, 44, 20, 20, 92, 20, 20, 20, 44, 20, 91, 91, 15, 14, 16, 8 respectively.

The titanic domain was obtained from Dawson, Robert J. MacG (Dawson, et.al, 1995). This data set has 2201 instances described by 3 attributes + one class attribute. This set includes 711 instances of one class and 1490 instances of another class.

Further an attempt is made to suggest the suitability of the different algorithms of association rule mining to a given case study. In what follows now we discuss different case studies pertaining to breast cancer, larynx cancer gene sequences and other datasets by employing Apriori, PredictiveApriori and Tertius Algorithms respectively.

For immediate and easier reference each of the data set are assigned numbers and given below.

Table 1.1: Data set Number Association

Data Set	Number
Sunburn	1
Imaginary Disease	2
Contact Lenses	3
Larynx Cancer	4
Zoo	5
Monk 3	6
Monk	7
Monk 1	8
Breast Cancer	9
Soya beans	10
Titanic	11
Mushroom	12

EXPERIMENT ANALYSIS

Execution of Apriori Algorithm

Parameter Selection

The basis for selecting various parameters is:

- Minimum support is the percentage of task relevant data transactions for which a pattern is true. The lower bound for the minimum support has a default value set to 0.1.

- Confidence is the certainty measure for association rules. If A and B are sets of items in any transactions, confidence is the percentage of transactions containing A that also contains B in an association $A \Rightarrow$. In Weka by default it has a value set to 0.9.
- The number of cycles is the number of iterations taken to generate the best rules. iv. The default number of best rules is kept at 1000.

Table 1.2: Execution of Apriori Algorithm

Sl. No	No. of instances	No. of attributes	Run Information					
			Min support	Min confidence	No. of cycles	No. of large frequent item sets L	Avg. size of L	No. of best rules found
1.	8	5	0.13	0.9	18	5	36	455
2.	10	5	0.1	0.9	18	5	19	237
1.	24	5	0.1	0.9	18	4	42	83
4.	41	2	0.1	0.9	18	2	3	1
5.	101	18	0.4	0.9	12	8	90	1000
6.	122	7	0.1	0.9	18	4	50	19
7.	124	7	0.1	0.9	18	4	50	11
8.	169	7	0.1	0.9	18	3	55	1
9.	286	10	0.5	0.9	10	4	4	536
10.	683	36	0.7	0.9	6	6	37	1000
11.	2201	4	0.1	0.9	18	4	9	24
12.	8124	23	0.45	0.9	111	6	55	1000

OBSERVATIONS

The above table reveals that while using Apriori Algorithm to different case studies :-

- As the minimum support reduces, the number of cycles increases.
- In general, the average size of large item sets depends on the minimum support.
- The number of best rules found for any data set is independent of the number of instances and attributes but generally depends on the minimum support.
- The number of cycles needed to generate the best rules is independent of the number of instances and attributes.
- The number and size of frequent itemsets generated by strong association rule in Apriori decreases with the increase in datasets.
- When the minimum support is low and the number of cycles taken is more the number of best rules generated are always not equal to the default value.
- When the number of attributes is above 15(see Sl. No.5, 10 and 12 in Table 1.2) the number of best rules obtained is as per the default value.
- In other cases, when the number of attributes is small the number of best rules are not obtained according to default value.

EXECUTION OF PREDICTIVE APRIORI ALGORITHM

Parameter Selection

The basis for selecting various parameters is:

- The run information for this algorithm is independent of the minimum support and confidence value for different datasets.

- The number of rules to be found is kept at 10000

Table 1.3: Execution of Predictive Apriori Algorithm

Sl. No	No. of instances	No. of attributes	Run Information / No. of best rules found
1.	8	5	44
2.	10	5	100
1.	24	5	242
4.	41	2	8
5.	101	18	10000
6.	122	7	6521
7.	124	7	9745
8.	169	7	6767
9.	286	10	10000
10.	683	36	10000
11.	2201	4	331
12.	8124	23	10

OBSERVATIONS

The above table reveals that while using Predictive Apriori Algorithm to different case studies

- The default value of 10000 rules is not found for all case studies which imply that the best rules found depend on the data set.
- A general rule cannot be formed for this algorithm as formed in Apriori algorithm.
- It returns the 'n' rules that maximize the expected accuracy where n is the number of best rules (see Sl. No.5, 9 and 10 in Table 1.3).
- In most cases the number of attributes being selected governs the number of best rules selected. The rules formation does not consider strange instances and may be termed as outliers.

EXECUTION OF TERTIUS ALGORITHM

Parameter Selection

This algorithm finds the rule according to the confirmation measures. It uses first order logic representation and includes various option like class Index, classification, confirmation Threshold, confirmation Values, frequency Threshold, horn Clauses, missing Values, negation, noise Threshold, number Literals, repeat Literals, roc Analysis, values Output.

The basis for selecting various parameters is:

- The number of rules generated depends on the number of hypotheses considered.
- The number of hypotheses explored is the number of rules that were "potentially interesting" and were considered for adding to the results or refining. This corresponds to the number of rules taken from the agenda in the algorithm.
- The run information for this algorithm is different from the other association algorithms.
- By default, the number of confirmation value is 1000 which gives the number of hypothesis considered and explored.
- The number of best rules found is also dependent on the number of confirmation value.

Table 1.4: Execution of Tertius Algorithm

Sl. No	# of instances	# of attributes	Run Information		
			# of hypotheses considered	# of hypotheses explored	# of best rules found
1.	8	5	3127	3127	299
2.	10	5	1596	1596	80
1.	24	5	2760	2760	894
4.	41	2	16	16	6
5.	101	18	4657888	2259393	4911
6.	122	7	29053	28510	1556
7.	124	7	29094	28546	1520
8.	169	7	29298	29178	1389
9.	286	10	514471	327637	1199
10.	683	36	7847055	2969128	1439
11.	2201	4	868	868	128
12.	8124	23	8179491	3787131	1914

OBSERVATION

The above table reveals the following information while using Tertius Algorithm to different case studies

- For non-numeric datasets as the number of attributes increases the number of hypotheses to be considered also increases.
- Though the number of hypothesis considered for numeric dataset is more the number of best rules formed is not more than 5000 for any data set.
- When the number of hypothesis is too large they are not fully explored.

Table 1.5: Comparative analysis from all the three algorithms

Sl. No	# of instances	# of attributes	# of best rules found		
			Apriori Algorithm	Predictive Apriori Algorithm	Tertius Algorithm
1.	8	5	455	44	299
2.	10	5	237	100	80
1.	24	5	83	242	894
4.	41	2	1	8	6
5.	101	18	1000	10000	4911
6.	122	7	19	6521	1556
7.	124	7	11	9745	1520
8.	169	7	1	6767	1389
9.	286	10	536	10000	1199
10.	683	36	24	331	1439
11.	2201	4	1000	10	128
12.	8124	23	455	44	1914

OBSERVATIONS OF ALL THREE ALGORITHMS

- The default value kept cannot be attained for all case studies for all the three algorithms.
- The number of best rules required depends on the number of attributes dataset.
- The datasets do not necessarily give the default number of rules for all the algorithms. iv.

When the number of attributes are less the number of best rules formed is less with all the three algorithms.

RESULTS AND DISCUSSIONS

The case studies considered in this paper are analyzed and executed from the Weka software to generate strong association rules using candidate generation with the different association rule mining algorithms and the findings are as below.

Suppose D denotes the data set and t denotes the time and i denotes the items that are generated then,

- As the minimum support reduces the number of cycles increases as shown in Table 1.2. ii. In general, the average size of large item sets depends on the minimum support.
- The number of best rules found for any data set is independent of the number of instances and attributes but generally depends on the minimum support.
- The number of cycles needed to generate the best rules is independent of the number of instances and attributes.
- During the execution it has been observed that the execution time of Apriori algorithm increases as the data set increases i.e. D α t.
- It has also been observed that the number and size of frequent itemsets generated by strong association rule in Apriori decreases with the increase in datasets i.e. D α 1/i.
- As the number of dataset increases Apriori requires more memory that leads to space complexity.
- Apriori may need to generate a huge number of candidate sets.
- It may need to repeatedly scan the database and check a large set of candidates by pattern matching. This is especially the case for mining long patterns.
- When the minimum confidence is low and the number of cycles taken is more the number of best rules generated are always not equal to the default value.
- The PredictiveApriori returns the 'n' number of best rules that maximize the expected accuracy.
- In general, the number of attributes being selected governs the number of best rules selected.
- The user has the option to specify how many rules have to be presented in PredictiveApriori and this is a more natural parameter to be used than minimum support and minimum confidence used by Apriori.
- The Predictive Apriori algorithm checks for redundancies.
- During execution it has been observed that the Predictive Apriori algorithm has a favourable computational performance since it uses dynamic pruning technique.
- In the Tertius algorithm the number of hypothesis considered and explored are directly proportional to the number of attributes and the number of instances.
- None of the three algorithms can handle numeric data. They can only use nominal attributes and datasets with numeric attributes have to be first discretized using the Weka toolkit. When discretization is done, the number of nominal values should be kept low otherwise the search space might be too wide for the search to work effectively.
- The model proposed for this mining is as shown in Figure 1.4 and is self-explanatory.

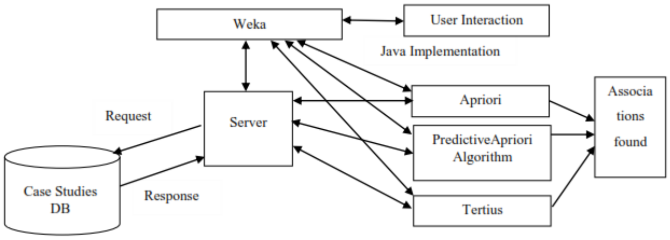


Fig 1.4 - Architecture for implementing the Association rules on Datasets using Weka

CONCLUSION

The Apriori Algorithm is the simplest algorithm to be used in association rule mining but the efficiency of the algorithm does not match with the size of the database to be analyzed. The Apriori Algorithm iteratively reduces the minimum support until it finds the required number of rules with the given minimum confidence. This increases the execution time and size of the frequent item sets. In the Predictive Apriori Algorithm a rule is added if the expected predictive accuracy of the particular rule is among the ‘n’ number of best rues and it is not a part of another rule with at least the same expected predictive accuracy. It searches with an increasing support threshold for the best 'n' rules concerning a support-based corrected confidence value. In Tertius Algorithm the search starts with an empty rule. The refinement operation is just adding an attribute-value pair either in the head or in the body of the rule. The algorithm is very slow and ways of making it run faster has to be considered.

REFERENCES

- A. Chandrakasan, H. Balakrishnan, “Energy efficient communication protocol for wireless micro sensor networks”, in: Hawaii International Conference on System Sciences (HICSS), 2016.
- A.E. Kamal, “Routing techniques in wireless sensor networks: a survey”, IEEE Wireless Communications, 11:6, 6–28, 2017.
- D. Estrin, “Scale: A tool for simple connectivity assessment in lossy environments”, Tech. Rep. 21, Center for Embedded Networked Sensing, University of California, Los Angeles, CA, USA, 2001.
- Kamal, A.E., "Routing techniques in wireless sensor networks: a survey", Wireless Communications, IEEE, vol.11, no.6, pp. 6-28, Dec. 2015.
- A. Lekha., "Efficiency of Data Mining Algorithms for large Biological Databases" ,Faculty of Computer Applications, Dr. M.G.R. Educational and Research Institute University, Chennai, 2014.
- P. Bates, “Debugging heterogeneous distributed systems using event based models of behavior”, ACM Transactions on Computer Systems, 13: 1, 2020.
- Minubhai Chaudari, Jigar Varmora,”Advance Privacy Preserving in Association rule Mining”,in IEEE conference (ICEEOT) 2016 .
- Sunita B. Aher, Lobo L.M.R.J, "A Comparative Study of Association Rule Algorithms for Course Recommender System in E-learning " International Journal of Computer Applications (0975 – 8887) Volume 39– No.1, February 2012.
- Tapan Sirole & Jaytrilok Choudhary, "A Survey of Various Methodologies for Hiding Sensitive Association Rules" International Journal of Computer Applications (0975 - 8887) Volume 96-No.18, June 2014.

- Qinbin Chen; Jia-yi Liu; Ke-ping Long, "A New Energy-Aware Routing Protocol for Wireless Sensor Networks", International Conference on Wireless Communications, Networking and Mobile Computing (WiCom 2007), pp.2444-2447, 21-25 Sept. 2017.
- R. Govindan,, "Understanding packet delivery performance in dense wireless sensor networks", in: Proceedings of the 1st ACM International Conference on Embedded Networked Sensor Systems, SENSYS, ACM Press, Los Angeles, CA, USA, 2019.
- Xiao Debao, "Mobile agent-based policy management for wireless sensor networks," Proceedings of 2005 International Conference on Wireless Communications, Networking and Mobile Computing, vol.2, pp. 1207-1210, 23-26 Sept. 2020.