Association rule learning

Association rule learning is a rule-based machine learning method for discovering interesting relations between variables in large databases. It is intended to identify strong rules discovered in databases using some measures of interestingness.[1]

Based on the concept of strong rules, Rakesh Agrawal, Tomasz Imieliński and Arun Swami[2] introduced association rules for discovering regularities between products in large-scale transaction data recorded by point-of-sale (POS) systems in supermarkets. For example, the rule $\{onions, potatoes\} \Rightarrow \{burger\}$ found in the sales data of a supermarket would indicate that if a customer buys onions and potatoes together, they are likely to also buy hamburger meat. Such information can be used as the basis for decisions about marketing activities such as, e.g., promotional pricing or product placements.

In addition to the above example from market basket analysis association rules are employed today in many application areas including Web usage mining, intrusion detection, continuous production, and bioinformatics. In contrast with sequence mining, association rule learning typically does not consider the order of items either within a transaction or across transactions.

Contents

Definition

Following the original definition by Agrawal, Imieliński, Swami[2] the problem of association rule mining is defined as:

Let $I = \{i_1, i_2, \ldots, i_n\}$ be a set of n binary attributes called *items.*

Let $D = \{t_1, t_2, \ldots, t_m\}$ be a set of transactions called the *database.*

Each *transaction* in D has a unique transaction ID and contains a subset of the items in I.

A *rule* is defined as an implication of the form:

$X \Rightarrow Y$, where $X, Y \subseteq I$.

In Agrawal, Imieliński, Swami[2] a *rule* is defined only between a set and a single item, $X \Rightarrow i_j$ for $i_j \in I$.

Every rule is composed by two different sets of items, also known as *itemsets,* X and Y, where X is called *antecedent* or left-hand-side (LHS) and Y *consequent* or right-hand-side (RHS).

To illustrate the concepts, we use a small example from the supermarket domain. The set of items is $I = \{\text{milk}, \text{bread}, \text{butter}, \text{beer}, \text{diapers}\}$ and in the table is shown a small database containing the items, where, in each entry, the value 1 means the presence of the item in the corresponding transaction, and the value 0 represents the absence of an item in that transaction.

An example rule for the supermarket could be $\{\text{butter}, \text{bread}\} \Rightarrow \{\text{milk}\}$ meaning that if butter and bread are bought, customers also buy milk.

Note: this example is extremely small. In practical applications, a rule needs a support of several hundred transactions before it can be considered statistically significant, and datasets often contain thousands or millions of transactions.

Example database with 5 transactions and 5 items

transaction ID	milk	bread	butter	beer	diapers
1	1	1	0	0	0
2	0	0	1	0	0
3	0	0	0	1	1
4	1	1	1	0	0
5	0	1	0	0	0

Useful Concepts

In order to select interesting rules from the set of all possible rules, constraints on various measures of significance and interest are used. The best-known constraints are minimum thresholds on support and confidence.

Let X be an itemset, $X \Rightarrow Y$ an association rule and T a set of transactions of a given database.

Support

Support is an indication of how frequently the itemset appears in the dataset.

The support of X with respect to T is defined as the proportion of transactions t in the dataset which contains the itemset X.

$$\text{supp}(X) = \frac{|\{t \in T; X \subseteq t\}|}{|T|}$$

In the example dataset, the itemset $X = \{\text{beer}, \text{diapers}\}$ has a support of $1/5 = 0.2$ since it occurs in 20% of all transactions (1 out of 5 transactions). The argument of $\text{supp}()$ is a set of preconditions, and thus becomes more restrictive as it grows (instead of more inclusive).[3]

Confidence

Confidence is an indication of how often the rule has been found to be true.

The *confidence* value of a rule, $X \Rightarrow Y$, with respect to a set of transactions T, is the proportion of the transactions that contains X which also contains Y.

Confidence is defined as:

$$\mathrm{conf}(X \Rightarrow Y) = \mathrm{supp}(X \cup Y)/\mathrm{supp}(X)$$

For example, the rule $\{\mathbf{butter}, \mathbf{bread}\} \Rightarrow \{\mathbf{milk}\}$ has a confidence of $0.2/0.2 = 1.0$ in the database, which means that for 100% of the transactions containing butter and bread the rule is correct (100% of the times a customer buys butter and bread, milk is bought as well).

Note that $\mathbf{supp}(X \cup Y)$ means the support of the union of the items in X and Y. This is somewhat confusing since we normally think in terms of probabilities of events and not sets of items. We can rewrite $\mathbf{supp}(X \cup Y)$ as the probability $P(E_X \cap E_Y)$, where E_X and E_Y are the events that a transaction contains itemset X and Y, respectively.[4]

Thus confidence can be interpreted as an estimate of the conditional probability $P(E_Y|E_X)$, the probability of finding the RHS of the rule in transactions under the condition that these transactions also contain the LHS.[3][5]

Lift

The *lift* of a rule is defined as:

$$\mathrm{lift}(X \Rightarrow Y) = \frac{\mathrm{supp}(X \cup Y)}{\mathrm{supp}(X) \times \mathrm{supp}(Y)}$$

or the ratio of the observed support to that expected if X and Y were independent.

For example, the rule $\{\mathbf{milk}, \mathbf{bread}\} \Rightarrow \{\mathbf{butter}\}$ has a lift of $\dfrac{0.2}{0.4 \times 0.4} = 1.25$.

If the rule had a lift of 1, it would imply that the probability of occurrence of the antecedent and that of the consequent are independent of each other. When two events are independent of each other, no rule can be drawn involving those two events.

If the lift is > 1, that lets us know the degree to which those two occurrences are dependent on one another, and makes those rules potentially useful for predicting the consequent in future data sets.

The value of lift is that it considers both the confidence of the rule and the overall data set.[3]

Conviction

The *conviction* of a rule is defined as $\mathbf{conv}(X \Rightarrow Y) = \dfrac{1 - \mathrm{supp}(Y)}{1 - \mathrm{conf}(X \Rightarrow Y)}$.

For example, the rule $\{\mathbf{milk}, \mathbf{bread}\} \Rightarrow \{\mathbf{butter}\}$ has a conviction of $\dfrac{1 - 0.4}{1 - 0.5} = 1.2$, and can be interpreted as the ratio of the expected frequency that X occurs without Y (that is to say, the frequency that the rule makes an incorrect prediction) if X and Y were independent divided by the observed frequency of incorrect predictions. In this example, the conviction value of 1.2 shows that the rule $\{\mathbf{milk}, \mathbf{bread}\} \Rightarrow \{\mathbf{butter}\}$ would be correct 20% more often (1.2 times as often) if the association between X and Y was purely random chance.

Process

Association rules are usually required to satisfy a user-specified minimum support and a user-specified minimum confidence at the same time. Association rule generation is usually split up into two separate steps:

1. A minimum support threshold is applied to find all *frequent itemsets* in a database.
2. A minimum confidence constraint is applied to these frequent itemsets in order to form rules.

While the second step is straightforward, the first step needs more attention.

Finding all frequent itemsets in a database is difficult since it involves searching all possible itemsets (item combinations). The set of possible itemsets is the power set over I and has size $2^n - 1$ (excluding the empty set which is not a valid itemset). Although the size of the power-set grows exponentially in the number of items n in I, efficient search is possible using the **downward-closure property** of support[2][6] (also called *anti-monotonicity*[7]) which guarantees that for a frequent itemset, all its subsets are also frequent and thus no infrequent itemset can be a subset of a frequent itemset. Exploiting this property, efficient algorithms (e.g., Apriori[8] and Eclat[9]) can find all frequent itemsets.
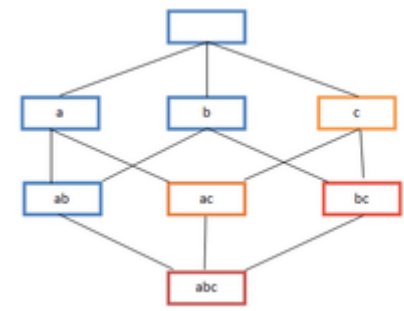
Frequent itemset lattice, where the color of the box indicates how many transactions contain the combination of items. Note that lower levels of the lattice can contain at most the minimum number of their parents' items; e.g. {ac} can have only at most $min(a, c)$ items. This is called the *downward-closure property*[2]

History

The concept of association rules was popularised particularly due to the 1993 article of Agrawal et al.,[2] which has acquired more than 18,000 citations according to Google Scholar, as of August 2015, and is thus one of the most cited papers in the Data Mining field. However, what is now called "association rules" is introduced already in the 1966 paper[10] on GUHA, a general data mining method developed by Petr Hájek et al.[11]

An early (circa 1989) use of minimum support and confidence to find all association rules is the Feature Based Modeling framework, which found all rules with $supp(X)$ and $conf(X \Rightarrow Y)$ greater than user defined constraints.[12]

Alternative measures of interestingness

In addition to confidence, other measures of *interestingness* for rules have been proposed. Some popular measures are:

- All-confidence[13]
- Collective strength[14]
- Conviction[15]
- Leverage[16]
- Lift (originally called interest)[17]

Several more measures are presented and compared by Tan et al.[18] and by Hahsler.[4] Looking for techniques that can model what the user has known (and using these models as interestingness measures) is currently an active research trend under the name of "Subjective Interestingness."

Statistically sound associations

One limitation of the standard approach to discovering associations is that by searching massive numbers of possible associations to look for collections of items that appear to be associated, there is a large risk of finding many spurious associations. These are collections of items that co-occur with unexpected frequency in the data, but only do so by chance. For example, suppose we are considering a collection of 10,000 items and looking for rules containing two items in the left-hand-side and 1 item in the right-hand-side. There are approximately 1,000,000,000,000 such rules. If we apply a statistical test for independence with a significance level

of 0.05 it means there is only a 5% chance of accepting a rule if there is no association. If we assume there are no associations, we should nonetheless expect to find 50,000,000,000 rules. Statistically sound association discovery[19][20] controls this risk, in most cases reducing the risk of finding *any* spurious associations to a user-specified significance level.

Algorithms

Many algorithms for generating association rules have been proposed.

Some well-known algorithms are Apriori, Eclat and FP-Growth, but they only do half the job, since they are algorithms for mining frequent itemsets. Another step needs to be done after to generate rules from frequent itemsets found in a database.

Apriori algorithm

Apriori[8] uses a breadth-first search strategy to count the support of itemsets and uses a candidate generation function which exploits the downward closure property of support.

Eclat algorithm

Eclat[9] (alt. ECLAT, stands for Equivalence Class Transformation) is a depth-first search algorithm using set intersection. It is a naturally elegant algorithm suitable for both sequential as well as parallel execution with locality-enhancing properties. It was first introduced by Zaki, Parthasarathy, Li and Ogihara in a series of papers written in 1997.[21][22]

FP-growth algorithm

FP stands for frequent pattern.[23]

In the first pass, the algorithm counts occurrence of items (attribute-value pairs) in the dataset, and stores them to 'header table'. In the second pass, it builds the FP-tree structure by inserting instances. Items in each instance have to be sorted by descending order of their frequency in the dataset, so that the tree can be processed quickly. Items in each instance that do not meet minimum coverage threshold are discarded. If many instances share most frequent items, FP-tree provides high compression close to tree root.

Recursive processing of this compressed version of main dataset grows large item sets directly, instead of generating candidate items and testing them against the entire database. Growth starts from the bottom of the header table (having longest branches), by finding all instances matching given condition. New tree is created, with counts projected from the original tree corresponding to the set of instances that are conditional on the attribute, with each node getting sum of its children counts. Recursive growth ends when no individual items conditional on the attribute meet minimum support threshold, and processing continues on the remaining header items of the original FP-tree.

Once the recursive process has completed, all large item sets with minimum coverage have been found, and association rule creation begins.[24]

Others

AprioriDP

AprioriDP[25] utilizes Dynamic Programming in Frequent itemset mining. The working principle is to eliminate the candidate generation like FP-tree, but it stores support count in specialized data structure instead of tree.

Context Based Association Rule Mining Algorithm

CBPNARM is an algorithm, developed in 2013, to mine association rules on the basis of context. It uses context variable on the basis of which the support of an itemset is changed on the basis of which the rules are finally populated to the rule set.

Node-set-based algorithms

FIN,[26] PrePost[27] and PPV[28] are three algorithms based on node sets. They use nodes in a coding FP-tree to represent itemsets, and employ a depth-first search strategy to discovery frequent itemsets using "intersection" of node sets.

GUHA procedure ASSOC

GUHA is a general method for exploratory data analysis that has theoretical foundations in observational calculi[29]

The ASSOC procedure[30] is a GUHA method which mines for generalized association rules using fast bitstrings operations. The association rules mined by this method are more general than those output by apriori, for example "items" can be connected both wit conjunction and disjunctions and the relation between antecedent and consequent of the rule is not restricted to setting minimum support and confidence as in apriori: an arbitrary combination of supported interest measures can be used.

OPUS search

OPUS is an efficient algorithm for rule discovery that, in contrast to most alternatives, does not require either monotone or anti-monotone constraints such as minimum support.[31] Initially used to find rules for a fixed consequent[31][32] it has subsequently been extended to find rules with any item as a consequent.[33] OPUS search is the core technology in the popular Magnum Opus association discovery system.

Lore

A famous story about association rule mining is the "beer and diaper" story. A purported survey of behavior of supermarket shoppers discovered that customers (presumably young men) who buy diapers tend also to buy beer. This anecdote became popular as an example of how unexpected association rules might be found from everyday data. There are varying opinions as to how much of the story is true.[34] Daniel Powers says:[34]

> In 1992, Thomas Blischok, manager of a retail consulting group at Teradata, and his staff prepared an analysis of 1.2 million market baskets from about 25 Osco Drug stores. Database queries were developed to identify affinities. The analysis "did discover that between 5:00 and 7:00 p.m. that consumers bought beer and diapers". Osco managers did NOT exploit the beer and diapers relationship by moving the products closer together on the shelves.

Other types of association rule mining

Multi-Relation Association Rules: Multi-Relation Association Rules (MRAR) are association rules where each item may have several relations. These relations indicate indirect relationship between the entities. Consider the following MRAR where the first item consists of three relations *live in, nearby* and *humid*: "Those who *live in* a place which is *nearby* a city with *humid* climate type and also are *younger* than 20 -> their *health condition* is good". Such association rules are extractable from RDBMS data or semantic web data.[35]

Context Based Association Rules are a form of association rule. **Context Based Association Rules** claims more accuracy in association rule mining by considering a hidden variable named context variable which changes the final set of association rules depending upon the value of context variables. For example, the baskets orientation in market basket analysis reflects an odd pattern in the early days of month. This might be because of abnormal context i.e. salary is drawn at the start of the month.[36]

Contrast set learning is a form of associative learning. **Contrast set learners** use rules that differ meaningfully in their distribution across subsets.[37][38]

Weighted class learning is another form of associative learning in which weight may be assigned to classes to give focus to a particular issue of concern for the consumer of the data mining results.

High-order pattern discovery facilitate the capture of high-order (polythetic) patterns or event associations that are intrinsic to complex real-world data.[39]

K-optimal pattern discovery provides an alternative to the standard approach to association rule learning that requires that each pattern appear frequently in the data.

Approximate Frequent Itemset mining is a relaxed version of Frequent Itemset mining that allows some of the items in some of the rows to be 0.[40]

Generalized Association Rules hierarchical taxonomy (concept hierarchy)

Quantitative Association Rules categorical and quantitative data[41][42]

Interval Data Association Rules e.g. partition the age into 5-year-increment ranged

Sequential pattern mining discovers subsequences that are common to more than minsup sequences in a sequence database, where minsup is set by the user. A sequence is an ordered list of transactions.[43]

Subspace Clustering a specific type of Clustering high-dimensional data, is in many variants also based on the downward-closure property for specific clustering models.[44]

Warmr is shipped as part of the ACE data mining suite. It allows association rule learning for first order relational rules.[45]

See also

- Sequence mining
- Production system (computer science)
- Learning classifier system
- Rule-based machine learning

References

1. Piatetsky-Shapiro, Gregory (1991), *Discovery, analysis, and presentation of strong rules*, in Piatetsky-Shapiro, Gregory; and Frawley, William J.; eds., *Knowledge Discovery in Databases*, AAAI/MIT Press, Cambridge, MA.

2. Agrawal, R.; Imieliński, T.; Swami, A. (1993). "Mining association rules between sets of items in large databases". *Proceedings of the 1993 ACM SIGMOD international conference on Management of data - SIGMOD '93*. p. 207. doi:10.1145/170035.170072 (https://doi.org/10.1145%2F170035.170072). ISBN 0897915925.

3. Hahsler, Michael (2005). "Introduction to arules – A computational environment for mining association rules and frequent item sets" (https://mran.revolutionanalytics.com/web/packages/arules/vignettes/arules.pdf) (PDF). *Journal of Statistical Software*.

4. Michael Hahsler (2015). A Probabilistic Comparison of Commonly Used Interest Measures for Association Rules. http://michael.hahsler.net/research/association_rules/measures.html

5. Hipp, J.; Güntzer, U.; Nakhaeizadeh, G. (2000). "Algorithms for association rule mining --- a general survey and comparison". *ACM SIGKDD Explorations Newsletter* **2**: 58. doi:10.1145/360402.360421 (https://doi.org/10.1145%2F360402.360421).

6. Tan, Pang-Ning; Michael, Steinbach; Kumar, Vipin (2005). "Chapter 6. Association Analysis: Basic Concepts and Algorithms" (http://www-users.cs.umn.edu/~kumar/dmbook/ch6.pdf) (PDF). *Introduction to Data Mining*. Addison-Wesley. ISBN 0-321-32136-7.

7. Pei, Jian; Han, Jiawei; and Lakshmanan, Laks VS.; *Mining frequent itemsets with convertible constraints*, in *Proceedings of the 17th International Conference on Data Engineering, April 2–6, 2001, Heidelberg, Germany*, 2001, pages 433-442

8. Agrawal, Rakesh; and Srikant, Ramakrishnan; *Fast algorithms for mining association rules in large databases* (http://rakesh.agrawal-family.com/papers/vldb94apriori.pdf), in Bocca, Jorge B.; Jarke, Matthias; and Zaniolo, Carlo; editors, *Proceedings of the 20th International Conference on Very Large Data Bases (VLDB), Santiago, Chile, September 1994*, pages 487-499

9. Zaki, M. J. (2000). "Scalable algorithms for association mining". *IEEE Transactions on Knowledge and Data Engineering*. **12** (3): 372–390. doi:10.1109/69.846291 (https://doi.org/10.1109%2F69.846291)

10. Hájek, Petr; Havel, Ivan; Chytil, Metoděj; *The GUHA method of automatic hypotheses determination*, Computing 1 (1966) 293-308

11. Hájek, Petr; Feglar, Tomas; Rauch, Jan; and Coufal, David; *The GUHA method, data preprocessing and mining*, Database Support for Data Mining Applications, Springer, 2004, ISBN 978-3-540-22479-2

12. Webb, Geoffrey (1989). "A Machine Learning Approach to Student Modelling". *Proceedings of the Third Australian Joint Conference on Artificial Intelligence (AI 89)*: 195–205.

13. Omiecinski, Edward R.; *Alternative interest measures for mining associations in databases*, IEEE Transactions on Knowledge and Data Engineering, 15(1):57-69, Jan/Feb 2003

14. Aggarwal, Charu C.; and Yu, Philip S.; *A new framework for itemset generation*, in *PODS 98, Symposium on Principles of Database Systems, Seattle, WA, USA, 1998*, pages 18-24

15. Brin, Sergey; Motwani, Rajeev; Ullman, Jeffrey D.; and Tsur, Shalom; *Dynamic itemset counting and implication rules for market basket data*, in *SIGMOD 1997, Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD 1997), Tucson, Arizona, USA, May 1997*, pp. 255-264

16. Piatetsky-Shapiro, Gregory; *Discovery, analysis, and presentation of strong rules*, Knowledge Discovery in Databases, 1991, pp. 229-248

17. Brin, Sergey; Motwani, Rajeev; Ullman, Jeffrey D.; and Tsur, Shalom; *Dynamic itemset counting and implication rules for market basket data*, in *SIGMOD 1997, Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD 1997), Tucson, Arizona, USA, May 1997*, pp. 265-276

18. Tan, Pang-Ning; Kumar, Vipin; and Srivastava, Jaideep; *Selecting the right objective measure for association analysis*, Information Systems, 29(4):293-313, 2004

19. Webb, Geoffrey I. (2007); *Discovering Significant Patterns*, Machine Learning 68(1), Netherlands: Springer, pp. 1-33 online access (https://link.springer.com/article/10.1007%2Fs10994-007-5006-x)

20. Gionis, Aristides; Mannila, Heikki; Mielikäinen, Taneli; and Tsaparas, Panayiotis; *Assessing Data Mining Results via Swap Randomization*, ACM Transactions on Knowledge Discovery from Data (TKDD), Volume 1, Issue 3 (December 2007), Article No. 14

21. Mohammed Javeed Zaki, Srinivasan Parthasarathy, M. Ogihara, Wei Li: New Algorithms for Fast Discovery of Association Rules. KDD 1997.

22. Mohammed Javeed Zaki, Srinivasan Parthasarathy, Mitsunori Ogihara, Wei Li: Parallel Algorithms for Discovery of Association Rules. Data Min. Knowl. Discov. 1(4): 343-373 (1997)

23. Han (2000). "Mining Frequent Patterns Without Candidate Generation". *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*. SIGMOD '00: 1–12. doi:10.1145/342009.335372 (https://doi.org/10.1145%2F342009.335372)

24. Witten, Frank, Hall: Data mining practical machine learning tools and techniques, 3rd edition

25. D. Bhalodiya, K. M. Patel and C. Patel. An Efficient way to Find Frequent Pattern with Dynamic Programming Approach [1] (http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=6780102&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D6780102). NIRMA UNIVERSITY INTERNATIONAL CONFERENCE ON ENGINEERING, NUiCONE-2013, 28-30 NOVEMBER, 2013.

26. Z. H. Deng and S. L. Lv. Fast mining frequent itemsets using Nodesets. [2] (http://www.sciencedirect.com/science/article/pii/S0957417414000463). Expert Systems with Applications, 41(10): 4505–4512, 2014.

27. Z. H. Deng, Z. Wang, and J. Jiang. A New Algorithm for Fast Mining Frequent Itemsets Using N-Lists. [3] (http://info.scichina.com:8084/sciFe/EN/abstract/abstract508369.shtml). SCIENCE CHINA Information Sciences, 55 (9): 2008 - 2030, 2012.

28. Z. H. Deng and Z. Wang. A New Fast Vertical Method for Mining Frequent Patterns. [4] (http://www.tandfonline.com/doi/abs/10.1080/18756891.2010.9727736). International Journal of Computational Intelligence Systems, 3(6): 733 - 744, 2010.

29. Rauch, Jan; *Logical calculi for knowledge discovery in databases*, in *Proceedings of the First European Symposium on Principles of Data Mining and Knowledge Discovery*, Springer, 1997, pp. 47-57

30. Hájek, Petr; Havránek, Tomáš (1978). *Mechanizing Hypothesis Formation: Mathematical Foundations for a General Theory* (http://www.cs.cas.cz/hajek/guhabook/) Springer-Verlag. ISBN 3-540-08738-9.

31. Webb, Geoffrey I. (1995); *OPUS: An Efficient Admissible Algorithm for Unordered Search*, Journal of Artificial Intelligence Research 3, Menlo Park, CA: AAAI Press, pp. 431-465 online access (http://webarchive.loc.gov/all/2001 1118141304/http://wwwcs.washington.edu/research/jair/abstracts/webb95a.html)

32. Bayardo, Roberto J., Jr; Agrawal, Rakesh; Gunopulos, Dimitrios (2000). "Constraint-based rule mining in large, dense databases". *Data Mining and Knowledge Discovery* **4** (2): 217–240. doi:10.1023/A:1009895914772(https://doi.org/10.1023%2FA%3A1009895914772)

33. Webb, Geoffrey I. (2000); *Efficient Search for Association Rules*, in Ramakrishnan, Raghu; and Stolfo, Sal; eds.; *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2000), Boston, MA*, New York, NY: The Association for Computing Machinery, pp. 99-107 online access (http://www.csse.monash.edu/~webb/Files/Webb00b.pdf)

34. http://www.dssresources.com/newsletters/66.php

35. Ramezani, Reza, Mohamad Saraee, and Mohammad Ali Nematbakhsh, *MRAR: Mining Multi-Relation Association Rules*, Journal of Computing and Security 1, no. 2 (2014)

36. Shaheen, M; Shahbaz, M; and Guergachi, A, *Context Based Positive and Negative Spatio-Temporal Association Rule Mining*, Elsevier Knowledge-Based Systems, Jan 2013, pp. 261-273

37. GI Webb and S. Butler and D. Newlands (2003). *On Detecting Differences Between Groups* (http://portal.acm.org/citation.cfm?id=956781) KDD'03 Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.

38. Menzies, Tim; and Hu, Ying; *Data Mining for Very Busy People*, IEEE Computer, October 2003, pp. 18-25

39. Wong, Andrew K.C.; Wang, Yang (1997). "High-order pattern discovery from discrete-valued data". *IEEE Transactions on Knowledge and Data Engineering (TKDE)*: 877–893.

40. Jinze Liu, Susan Paulsen, Xing Sun, Wei Wang, Andrew Nobel, J. P. (2006). Mining approximate frequent itemsets in the presence of noise: Algorithm and analysis. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.62.3805

41. Angiulli, Fabrizio; Ianni, Giovambattista; Palopoli, Luigi, "On the complexity of inducing categorical and quantitative association rules" (https://doi.org/10.1016/j.tcs.2003.12.017), *Theoretical Computer Science* **314** (1-2): 217–249. doi:10.1016/j.tcs.2003.12.017(https://doi.org/10.1016%2Fj.tcs.2003.12.017)

42. Salleb-Aouissi, Ansaf; Vrain, Christel; Nortet, Cyril (2007). "QuantMiner: A Genetic Algorithm for Mining Quantitative Association Rules". *International Joint Conference on Artificial Intelligence (IJCAI)*: 1035–1040.

43. Zaki, Mohammed J. (2001); *SPADE: An Efficient Algorithm for Mining Frequent Sequences*, Machine Learning Journal, 42, pp. 31–60

44. Zimek, Arthur; Assent, Ira; Veeken, Jilles (2014). "Frequent Pattern Mining Algorithms for Data Clustering": 403–423. doi:10.1007/978-3-319-07821-2_16(https://doi.org/10.1007%2F978-3-319-07821-2_16)

45. "Warmr: a data mining tool for chemical data". *J Comput Aided Mol Des* **15** (2): 173–81. Feb 2001. PMID 11272703 (https://www.ncbi.nlm.nih.gov/pubmed/11272703).

External links

Bibliographies

- Extensive Bibliography on Association Rules by J.M. Luna
- Annotated Bibliography on Association Rules by M. Hahsler
- Statsoft Electronic Statistics Textbook: Association Rules by Dell Software

Foundation, Inc., a non-profit organization.