GESTS International Transactions on Computer Science and Engineering, Vol.32 (1), 2006, pp. 71-82

Association Rules Mining: A Recent Overview

Sotiris Kotsiantis, Dimitris Kanellopoulos

Educational Software Development Laboratory
Department of Mathematics, University of Patras, Greece
sotos@math.upatras.gr, dkanellop@teipat.gr

Abstract. In this paper, we provide the preliminaries of basic concepts about association rule mining and survey the list of existing association rule mining techniques. Of course, a single article cannot be a complete review of all the algorithms, yet we hope that the references cited will cover the major theoretical issues, guiding the researcher in interesting research directions that have yet to be explored.

1 Introduction

Association rule mining, one of the most important and well researched techniques of data mining, was first introduced in [1]. It aims to extract interesting correlations, frequent patterns, associations or casual structures among sets of items in the transaction databases or other data repositories. Association rules are widely used in various areas such as telecommunication networks, market and risk management, inventory control etc. Various association mining techniques and algorithms will be briefly introduced and compared later.

Association rule mining is to find out association rules that satisfy the predefined minimum support and confidence from a given database. The problem is usually decomposed into two subproblems. One is to find those itemsets whose occurrences exceed a predefined threshold in the database; those itemsets are called frequent or large itemsets. The second problem is to generate association rules from those large itemsets with the constraints of minimal confidence. Suppose one of the large itemsets is L_k, $L_k = \{I_1, I_2, \dots, I_k\}$, association rules with this itemsets are generated in the following way: the first rule is $\{I_1, I_2, \dots, I_{k-1}\} \Rightarrow \{I_k\}$, by checking the confidence this rule can be determined as interesting or not. Then other rule are generated by deleting the last items in the antecedent and inserting it to the consequent, further the confidences of the new rules are checked to determine the interestingness of them. Those processes iterated until the antecedent becomes empty. Since the second subproblem is quite straight forward, most of the researches focus on the first subproblem.

The first sub-problem can be further divided into two sub-problems: candidate large itemsets generation process and frequent itemsets generation process. We call those itemsets whose support exceed the support threshold as large or frequent itemsets, those itemsets that are expected or have the hope to be large or frequent are called candidate itemsets.

In many cases, the algorithms generate an extremely large number of association rules, often in thousands or even millions. Further, the association rules are sometimes very large. It is nearly impossible for the end users to comprehend or validate such large number of complex association rules, thereby limiting the usefulness of the data mining results. Several strategies have been proposed to reduce the number of association rules, such as generating only “interesting” rules, generating only “non-redundant” rules, or generating only those rules satisfying certain other criteria such as coverage, leverage, lift or strength.

Hegland [16] reviews the most well known algorithm for producing association rules - Apriori and discuss variants for distributed data, inclusion of constraints and data taxonomies. The review ends with an outlook on tools which have the potential to deal with long itemsets and considerably reduce the amount of (uninteresting) itemsets returned.

In this paper, we surveyed the most recent existing association rule mining techniques. The organization of the rest of the paper is as follows. Section 2 provides the preliminaries of basic concepts and their notations to facilitate the discussion and describes the well known algorithms. Section 3 describes the methods that have been proposed for increasing the efficiency of association rules algorithms. Section 4 refers to the categories of databases in which association rule can be applied. Section 5 presents the recent advances in association rule discovery. Finally, Section 6 concludes the paper.

2 Basic Concepts & Basic Association Rules Algorithms

Let $I=I_1, I_2, \ldots, I_m$ be a set of m distinct attributes, T be transaction that contains a set of items such that $T \subseteq I$, D be a database with different transaction records Ts. An association rule is an implication in the form of $X \Rightarrow Y$, where $X, Y \subset I$ are sets of items called itemsets, and $X \cap Y = \varnothing$. X is called antecedent while Y is called consequent, the rule means X implies Y.

There are two important basic measures for association rules, support(s) and confidence(c). Since the database is large and users concern about only those frequently purchased items, usually thresholds of support and confidence are predefined by users to drop those rules that are not so interesting or useful. The two thresholds are called minimal support and minimal confidence respectively. Support(s) of an association rule is defined as the percentage/fraction of records that contain $X \cup Y$ to the total number of records in the database. Suppose the support of an item is 0.1%, it means only 0.1 percent of the transaction contain purchasing of this item.

Confidence of an association rule is defined as the percentage/fraction of the number of transactions that contain $X \cup Y$ to the total number of records that contain X. Confidence is a measure of strength of the association rules, suppose the confidence of the association rule $X \Rightarrow Y$ is 80%, it means that 80% of the transactions that contain X also contain Y together.

In general, a set of items (such as the antecedent or the consequent of a rule) is called an itemset. The number of items in an itemset is called the length of an itemset. Itemsets of some length k are referred to as k-itemsets.

Generally, an association rules mining algorithm contains the following steps:

- The set of candidate k-itemsets is generated by 1-extensions of the large (k -1)-itemsets generated in the previous iteration.
- Supports for the candidate k-itemsets are generated by a pass over the database.
- Itemsets that do not have the minimum support are discarded and the remaining itemsets are called large k-itemsets.

This process is repeated until no more large itemsets are found.

The AIS algorithm was the first algorithm proposed for mining association rule [1]. In this algorithm only one item consequent association rules are generated, which means that the consequent of those rules only contain one item, for example we only generate rules like $X \cap Y \Rightarrow Z$ but not those rules as $X \Rightarrow Y \cap Z$. The main drawback of the AIS algorithm is too many candidate itemsets that finally turned out to be small are generated, which requires more space and wastes much effort that turned out to be useless. At the same time this algorithm requires too many passes over the whole database.

Apriori is more efficient during the candidate generation process [2]. Apriori uses pruning techniques to avoid measuring certain itemsets, while guaranteeing completeness. These are the itemsets that the algorithm can prove will not turn out to be large. However there are two bottlenecks of the Apriori algorithm. One is the complex candidate generation process that uses most of the time, space and memory. Another bottleneck is the multiple scan of the database. Based on Apriori algorithm, many new algorithms were designed with some modifications or improvements.

3 Increasing the Efficiency of Association Rules Algorithms

The computational cost of association rules mining can be reduced in four ways:

- by reducing the number of passes over the database
- by sampling the database
- by adding extra constraints on the structure of patterns
- through parallelization.

In recent years much progress has been made in all these directions.

3.1 Reducing the number of passes over the database

FP-Tree [15], frequent pattern mining, is another milestone in the development of association rule mining, which breaks the main bottlenecks of the Apriori. The frequent itemsets are generated with only two passes over the database and without any candidate generation process. FP-tree is an extended prefix-tree structure storing crucial, quantitative information about frequent patterns. Only frequent length-1 items will have nodes in the tree, and the tree nodes are arranged in such a way that more frequently occurring nodes will have better chances of sharing nodes than less frequently occurring ones.

FP-Tree scales much better than Apriori because as the support threshold goes down, the number as well as the length of frequent itemsets increase dramatically. The candidate sets that Apriori must handle become extremely large, and the pattern matching with a lot of candidates by searching through the transactions becomes very expensive. The frequent patterns generation process includes two sub processes: constructing the FT-Tree, and generating frequent patterns from the FP-Tree. The mining result is the same with Apriori series algorithms.

To sum up, the efficiency of FP-Tree algorithm account for three reasons. First the FP-Tree is a compressed representation of the original database because only those frequent items are used to construct the tree, other irrelevant information are pruned. Secondly this algorithm only scans the database twice. Thirdly, FP-Tree uses a divide and conquer method that considerably reduced the size of the subsequent conditional FP-Tree. In [15] there are examples to illustrate all the detail of this mining process. Every algorithm has his limitations, for FP-Tree it is difficult to be used in an interactive mining system. During the interactive mining process, users may change he threshold of support according to the rules. However for FP-Tree the changing of support may lead to repetition of the whole mining process. Another limitation is that FP-Tree is that it is not suitable for incremental mining. Since as time goes on databases keep changing, new datasets may be inserted into the database, those insertions may also lead to a repetition of the whole process if we employ FP-Tree algorithm.

TreeProjection is another efficient algorithm recently proposed in [3]. The general idea of TreeProjection is that it constructs a lexicographical tree and projects a large database into a set of reduced, item-based sub-databases based on the frequent patterns mined so far. The number of nodes in its lexicographic tree is exactly that of the frequent itemsets. The efficiency of TreeProjection can be explained by two main factors: (1) the transaction projection limits the support counting in a relatively small space; and (2) the lexicographical tree facilitates the management and counting of candidates and provides the flexibility of picking efficient strategy during the tree generation and transaction projection phrases.

Wang and Tjortjis [38] presented PRICES, an efficient algorithm for mining association rules. Their approach reduces large itemset generation time, known to be the most time-consuming step, by scanning the database only once and using logical operations in the process. Another algorithm for efficient generating large frequent candidate sets is proposed by [36], which is called Matrix Algorithm. The algorithm generates a matrix which entries 1 or 0 by passing over the cruel database only once, and then the frequent candidate sets are obtained from the resulting matrix. Finally association rules are mined from the frequent candidate sets. Experiments results confirm that the proposed algorithm is more effective than Apriori Algorithm.

3.2 Sampling

Toivonen [33] presented an association rule mining algorithm using sampling. The approach can be divided into two phases. During phase 1 a sample of the database is obtained and all associations in the sample are found. These results are then validated against the entire database. To maximize the effectiveness of the overall approach, the author makes use of lowered minimum support on the sample. Since the approach is probabilistic (i.e. dependent on the sample containing all the relevant associations) not all the rules may be found in this first pass. Those associations that were deemed not frequent in the sample but were actually frequent in the entire dataset are used to construct the complete set of associations in phase 2.

Parthasarathy [24] presented an efficient method to progressively sample for association rules. His approach relies on a novel measure of model accuracy (self-similarity of associations across progressive samples), the identification of a representative class of frequent itemsets that mimic (extremely accurately) the self-similarity values across the entire set of associations, and an efficient sampling methodology that hides the overhead of obtaining progressive samples by overlapping it with useful computation.

Chuang et al. [11] explore another progressive sampling algorithm, called Sampling Error Estimation (SEE), which aims to identify an appropriate sample size for mining association rules. SEE has two advantages. First, SEE is highly efficient because an appropriate sample size can be determined without the need of executing association rules. Second, the identified sample size of SEE is very accurate, meaning that association rules can be highly efficiently executed on a sample of this size to obtain a sufficiently accurate result.

Especially, if data comes as a stream flowing at a faster rate than can be processed, sampling seems to be the only choice. How to sample the data and how big the sample size should be for a given error bound and confidence level are key issues for particular data mining tasks. Li and Gopalan [19] derive the sufficient sample size based on central limit theorem for sampling large datasets with replacement.

3.3 Parallelization

Association rule discovery techniques have gradually been adapted to parallel systems in order to take advantage of the higher speed and greater storage capacity that they offer [41]. The transition to a distributed memory system requires the partitioning of the database among the processors, a procedure that is generally carried out indiscriminately

Cheung et al. [9] presented an algorithm called FDM. FDM is a parallelization of Apriori to (shared nothing machines, each with its own partition of the database. At every level and on each machine, the database scan is performed independently on the local partition. Then a distributed pruning technique is employed. Schuster and Wolff [29] described another Apriori based D-ARM algorithm - DDM. As in FDM, candidates in DDM are generated levelwise and are then counted by each node in its local database. The nodes then perform a distributed decision protocol in order to find out which of the candidates are frequent and which are not.

Another efficient parallel algorithm FPM (Fast Parallel Mining) for mining association rules on a shared-nothing parallel system has been proposed by [10]. It adopts the count distribution approach and has incorporated two powerful candidate pruning techniques, i.e., distributed pruning and global pruning. It has a simple communication scheme which performs only one round of message exchange in each iteration. A new algorithm, Data Allocation Algorithm (DAA), is presented in [21] that uses Principal Component Analysis to improve the data distribution prior to FPM.

Parthasarathy et al. [23] have written an excellent recent survey on parallel association rule mining with sharedmemory architecture covering most trends, challenges and approaches adopted for parallel data mining. All approaches spelled out and compared in this extensive survey are apriori-based. More recently, Tang and Turkia [25] proposed a parallelization scheme which can be used to parallelize the efficient and fast frequent itemset mining algorithms based on FP-trees.

3.4 Constraints based association rule mining

Many data mining techniques consist in discovering patterns frequently occurring in the source dataset. Typically, the goal is to discover all the patterns whose frequency in the dataset exceeds a user-specified threshold. However, very often users want to restrict the set of patterns to be discovered by adding extra constraints on the structure of patterns. Data mining systems should be able to exploit such constraints to speed-up the mining process. Techniques applicable to constraint-driven pattern discovery can be classified into the following groups:

- post-processing (filtering out patterns that do not satisfy user-specified pattern constraints after the actual discovery process);
- pattern filtering (integration of pattern constraints into the actual mining process in order to generate only patterns satisfying the constraints);
- dataset filtering (restricting the source dataset to objects that can possibly contain patterns that satisfy pattern constraints).

Wojciechowski and Zakrzewicz [39] focus on improving the efficiency of constraint-based frequent pattern mining by using dataset filtering techniques. Dataset filtering conceptually transforms a given data mining task into an equivalent one operating on a smaller dataset. Tien Dung Do et al [14] proposed a specific type of constraints called category-based as well as the associated algorithm for constrained rule mining based on Apriori. The Category-based Apriori algorithm reduces the computational complexity of the mining process by bypassing most of the subsets of the final itemsets. An experiment has been conducted to show the efficiency of the proposed technique.

Rapid Association Rule Mining (RARM) [13] is an association rule mining method that uses the tree structure to represent the original database and avoids candidate generation process. In order to improve the efficiency of existing mining algorithms, constraints were applied during the mining process to generate only those association rules that are interesting to users instead of all the association rules.

4 Categories of Databases in which Association Rules are applied

Transactional database refers to the collection of transaction records, in most cases they are sales records. With the popularity of computer and e-commerce, massive transactional databases are available now. Data mining on transactional database focuses on the mining of association rules, finding the correlation between items in the transaction records.

One of data mining techniques, generalized association rule mining with taxonomy, is potential to discover more useful knowledge than ordinary flat association rule mining by taking application specific information into account [27]. In particular in retail one might consider as items particular brands of items or whole groups like milk, drinks or food. The more general the items chosen the higher one can expect the support to be. Thus one might be interested in discovering frequent itemsets composed of items which themselves form a taxonomy. Earlier work on mining generalized association rules ignore the fact that the taxonomies of items cannot be kept static while new transactions are continuously added into the original database. How to effectively update the discovered generalized association rules to reflect the database change with taxonomy evolution and transaction update is a crucial task. Tseng et al [34] examine this problem and propose a novel algorithm, called IDTE, which can incrementally update the discovered generalized association rules when the taxonomy of items is evolved with new transactions insertion to the database. Empirical evaluations show that this algorithm can maintain its performance even in large amounts of incremental transactions and high degree of taxonomy evolution, and is more than an order of magnitude faster than applying the best generalized associations mining algorithms to the whole updated database.

Spatial databases usually contain not only traditional data but also the location or geographic information about the corresponding data. Spatial association rules describe the relationship between one set of features and another set of features in a spatial database, for example (Most business centers in Greece are around City Hall), the spatial operations that used to describe the correlation can be within, near, next to, etc. The form of spatial association rules is also $X \Rightarrow Y$, where X, Y are sets of predicates and of which some are spatial predicates, and at least one must be a spatial predicate [30].

Temporal association rules can be more useful and informative than basic association rules. For example rather than the basic association rule {diapers}⇒{beer}, mining from the temporal data we can get a more insight rule that the support of {diapers}⇒{beer} jumps to 50% during 6pm to 9pm everyday, obviously retailers can make more efficient promotion strategy by using temporal association rule. In [35] an algorithm for mining periodical patterns and episode sequential patterns was introduced.

5 Recent advances in association rule discovery

A serious problem in association rule discovery is that the set of association rules can grow to be unwieldy as the number of transactions increases, especially if the support and confidence thresholds are small. As the number of frequent itemsets increases, the number of rules presented to the user typically increases proportionately. Many of these rules may be redundant.

5.1 Redundant Association Rules

To address the problem of rule redundancy, four types of research on mining association rules have been performed. First, rules have been extracted based on user-defined templates or item constraints [6]. Secondly, researchers have developed interestingness measures to select only interesting rules [17]. Thirdly, researchers have proposed inference rules or inference systems to prune redundant rules and thus present smaller, and usually more understandable sets of association rules to the user [12]. Finally, new frameworks for mining association rule have been proposed that find association rules with different formats or properties [8].

Ashrafi et al [4] presented several methods to eliminate redundant rules and to produce small number of rules from any given frequent or frequent closed itemsets generated. Ashrafi et al [5] present additional redundant rule elimination methods that first identify the rules that have similar meaning and then eliminate those rules. Furthermore, their methods eliminate redundant rules in such a way that they never drop any higher confidence or interesting rules from the resultant rule set.

Jaroszewicz and Simovici [18] presented another solution to the problem using the Maximum Entropy approach. The problem of efficiency of Maximum Entropy computations is addressed by using closed form solutions for the most frequent cases. Analytical and experimental evaluation of their proposed technique indicates that it efficiently produces small sets of interesting association rules.

Moreover, there is a need for human intervention in mining interesting association rules. Such intervention is most effective if the human analyst has a robust visualization tool for mining and visualizing association rules. Techapichetvanich and Datta [31] presented a three-step visualization method for mining market basket association rules. These steps include discovering frequent itemsets, mining association rules and finally visualizing the mined association rules.

5.2 Other measures as interestingness of an association

Omiecinski [22] concentrates on finding associations, but with a different slant. That is, he takes a different view of significance. Instead of support, he considers other measures, which he calls all-confidence, and bond. All these measures are indicators of the degree to which items in an association are related to each other. With all-confidence, an association is deemed interesting if all rules that can be produced from that association have a confidence greater than or equal to a minimum all-confidence value. Bond is another measure of the interestingness of an association. With regard to data mining, it is similar to support but with respect to a subset of the data rather than the entire data set. This has similarities to the work in [26] except in their work they define data subsets based on the data satisfying certain time constraints. The idea is to find all itemsets that are frequent in a set of user-defined time intervals. In this case, the characteristics of the data define the subsets not the end-user. Omiecinski [22] proved that if associations have a minimum all-confidence or minimum bond, then those associations will have a given lower bound on their minimum support and the rules produced from those associations will have a given lower bound on their minimum confidence as well. The performance results showed that the algorithm can find large itemsets efficiently.

In [8], the authors mine association rules that identify correlations and consider both the absence and presence of items as a basis for generating the rules. The measure of significance of associations that is used is the chi-squared test for correlation from classical statistics. In [7], the authors still use support as part of their measure of interest of an association. However, when rules are generated, instead of using confidence, the authors use a metric they call conviction, which is a measure of implication and not just co-occurrence.

In [20], the authors present an approach to the rare item problem. The dilemma that arises in the rare item problem is that searching for rules that involve infrequent (i.e., rare) items requires a low support but using a low support will typically generate many rules that are of no interest. Using a high support typically reduces the number of rules mined but will eliminate the rules with rare items. The authors attack this problem by allowing users to specify different minimum supports for the various items in their mining algorithm.

5.3 Negative Association Rules

Typical association rules consider only items enumerated in transactions. Such rules are referred to as positive association rules. Negative association rules also consider the same items, but in addition consider negated items (i.e. absent from transactions). Negative association rules are useful in market-basket analysis to identify products that conflict with each other or products that complement each other. Mining negative association rules is a difficult task, due to the fact that there are essential differences between positive and negative association rule mining. The researchers attack two key problems in negative association rule mining: (i) how to effectively search for interesting itemsets, and (ii) how to effectively identify negative association rules of interest.

Brin et. al [8] mentioned for the first time in the literature the notion of negative relationships. Their model is chi-square based. They use the statistical test to verify the independence between two variables. To determine the nature (positive or negative) of the relationship, a correlation metric was used. In [28] the authors present a new idea to mine strong negative rules. They combine positive frequent itemsets with domain knowledge in the form of taxonomy to mine negative associations. However, their algorithm is hard to generalize since it is domain dependant and requires a predefined taxonomy. A similar approach is described in [37]. Wu et al [40] derived a new algorithm for generating both positive and negative association rules. They add on top of the support-confidence framework another measure called *mininterest* for a better pruning of the frequent itemsets generated. In [32] the authors use only negative associations of the type $X \Rightarrow \neg Y$ to substitute items in market basket analysis.

6 Conclusion

Association rule mining has a wide range of applicability such market basket analysis, medical diagnosis/ research, Website navigation analysis, homeland security and so on. In this paper, we surveyed the list of existing association rule mining techniques. The conventional algorithm of association rules discovery proceeds in two steps. All frequent itemsets are found in the first step. The frequent itemset is the itemset that is included in at least *minsup* transactions. The association rules with the confidence at least *minconf* are generated in the second step.

End users of association rule mining tools encounter several well known problems in practice. First, the algorithms do not always return the results in a reasonable time. 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 itemsets 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 itemsets, let alone to generate rules, since they typically produce an exponential number of frequent itemsets; finding long itemsets of length 20 or 30 is not uncommon. Although several different strategies have been proposed to tackle efficiency issues, they are not always successful.

References

[1] Agrawal, R., Imielinski, T., and Swami, A. N. 1993. Mining association rules between sets of items in large databases. In Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, 207-216.

[2] Agrawal, R. and Srikant, R. 1994. Fast algorithms for mining association rules. In Proc. 20th Int. Conf. Very Large Data Bases, 487-499.

[3] Agarwal, R. Aggarwal, C. and Prasad V., A tree projection algorithm for generation of frequent itemsets. In J. Parallel and Distributed Computing, 2000.

[4] Ashrafi, M., Taniar, D. Smith, K. A New Approach of Eliminating Redundant Association Rules, Lecture Notes in Computer Science, Volume 3180, 2004, Pages 465 – 474

[5] Ashrafi, M., Taniar, D., Smith, K., Redundant Association Rules Reduction Techniques, Lecture Notes in Computer Science, Volume 3809, 2005, pp. 254 - 263

[6] Baralis, E., Psaila, G., Designing templates for mining association rules. Journal of Intelligent Information Systems, 9(1):7-32, July 1997.

[7] Brin, S., Motwani, R., Ullman, J. D., and Tsur, S. 1997. Dynamic itemset 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.

[8] Brin, S., Motwani, R. and Silverstein, C., ”Beyond Market Baskets: Generalizing Association Rules to Correlations,” Proc. ACM SIGMOD Conf., pp. 265-276, May 1997.

[9] Cheung, D., Han, J., Ng, V., Fu, A. and Fu, Y. (1996), A fast distributed algorithm for mining association rules, in `Proc. of 1996 Int'l. Conf. on Parallel and Distributed Information Systems', Miami Beach, Florida, pp. 31 - 44.

[10] Cheung, D., Xiao, Y., Effect of data skewness in parallel mining of association rules, Lecture Notes in Computer Science, Volume 1394, Aug 1998, Pages 48 - 60

[11] Chuang, K., Chen, M., Yang, W., Progressive Sampling for Association Rules Based on Sampling Error Estimation, Lecture Notes in Computer Science, Volume 3518, Jun 2005, Pages 505 - 515

[12] Cristofor, L., Simovici, D., Generating an informative cover for association rules. In Proc. of the IEEE International Conference on Data Mining, 2002.

[13] Das, A., Ng, W.-K., and Woon, Y.-K. 2001. Rapid association rule mining. In Proceedings of the tenth international conference on Information and knowledge management. ACM Press, 474-481.

[14] Tien Dung Do, Siu Cheung Hui, Alvis Fong, Mining Frequent Itemsets with Category-Based Constraints, Lecture Notes in Computer Science, Volume 2843, 2003, pp. 76 - 86

[15] Han, J. and Pei, J. 2000. Mining frequent patterns by pattern-growth: methodology and implications. ACM SIGKDD Explorations Newsletter 2, 2, 14-20.

[16] Hegland, M., Algorithms for Association Rules, Lecture Notes in Computer Science, Volume 2600, Jan 2003, Pages 226 - 234

[17] Hilderman R. J., Hamilton H. J., Knowledge Discovery and Interest Measures, Kluwer Academic, Boston, 2002.

[18] Jaroszewicz, S., Simovici, D., Pruning Redundant Association Rules Using Maximum Entropy Principle, Lecture Notes in Computer Science, Volume 2336, Jan 2002, pp 135-142

[19] Li, Y., Gopalan, R., Effective Sampling for Mining Association Rules, Lecture Notes in Computer Science, Volume 3339, Jan 2004, Pages 391 - 401

[20] Liu, B. Hsu, W., Ma, Y., ”Mining Association Rules with Multiple Minimum Supports,” Proc. Knowledge Discovery and Data Mining Conf., pp. 337-341, Aug. 1999.

[21] Manning, A., Keane, J., Data Allocation Algorithm for Parallel Association Rule Discovery, Lecture Notes in Computer Science, Volume 2035, Page 413-420.

[22] Omiecinski, E. (2003), Alternative Interest Measures for Mining Associations in Databases, IEEE Transactions on Knowledge and Data Engineering, Vol. 15, No. 1, pp. 57-69.

[23] Parthasarathy, S., Zaki, M. J., Ogihara, M., Parallel data mining for association rules on shared-memory systems. Knowledge and Information Systems: An International Journal, 3(1):1–29, February 2001.

[24] Parthasarathy, S., Efficient Progressive Sampling for Association Rules. ICDM 2002: 354-361.

[25] Tang, P., Turkia, M., Parallelizing frequent itemset mining with FP-trees. Technical Report titus.compsci.ualr.edu/~ptang/papers/par-fi.pdf, Department of Computer Science, University of Arkansas at Little Rock, 2005.

[26] Ramaswamy, S., Mahajan, S., Silbershatz, A., ”On the Discovery of Interesting Patterns in Association Rules,” Proc. Very Large Databases Conf., pp. 368-379, Sept. 1998.

[27] Sarawagi, S., Thomas, S., “Mining Generalized Association Rules and Sequential Patterns Using SQL Queries”. In Proc. of KDD Conference, 1998.

[28] Savasere, A., Omiecinski, E., Navathe, S.: Mining for strong negative associations in a large database of customer transactions. In: Proc. of ICDE. (1998) 494–502

[29] Schuster, A. and Wolff, R. (2001), Communication-efficient distributed mining of association rules, in `Proc. of the 2001 ACM SIGMOD Int'l. Conference on Management of Data', Santa Barbara, California, pp. 473-484.

[30] Sharma, L.K., Vyas, O.P., Tiwary, U.S., Vyas, R. A Novel Approach of Multilevel Positive and Negative Association Rule Mining for Spatial Databases, Lecture Notes in Computer Science, Volume 3587, Jul 2005, Pages 620 - 629
[31] Techapichetvanich, K., Datta, A., Visual Mining of Market Basket Association Rules, Lecture Notes in Computer Science, Volume 3046, Jan 2004, Pages 479 - 488
[32] Teng, W., Hsieh, M., Chen, M.: On the mining of substitution rules for statistically dependent items. In: Proc. of ICDM. (2002) 442–449
[33] Toivonen, H. (1996), Sampling large databases for association rules, in `The VLDB Journal', pp. 134-145.
[34] Tseng, M., Lin, W., Jeng, R., Maintenance of Generalized Association Rules Under Transaction Update and Taxonomy Evolution, Lecture Notes in Computer Science, Volume 3589, Sep 2005, Pages 336 – 345
[35] Verma, K., Vyas, O.P., Vyas, R., Temporal Approach to Association Rule Mining Using T-Tree and P-Tree, Lecture Notes in Computer Science, Volume 3587, Jul 2005, Pages 651 - 659
[36] Yuan, Y., Huang, T., A Matrix Algorithm for Mining Association Rules, Lecture Notes in Computer Science, Volume 3644, Sep 2005, Pages 370 - 379
[37] Yuan, X., Buckles, B., Yuan, Z., Zhang, J.: Mining negative association rules. In: Proc. of ISCC. (2002) 623–629
[38] Wang, C., Tjortjis, C., PRICES: An Efficient Algorithm for Mining Association Rules, Lecture Notes in Computer Science, Volume 3177, Jan 2004, Pages 352 - 358
[39] Wojciechowski, M., Zakrzewicz, M., Dataset Filtering Techniques in Constraint-Based Frequent Pattern Mining, Lecture Notes in Computer Science, Volume 2447, 2002, pp. 77-83
[40] Wu, X., Zhang, C., Zhang, S.: Efficient Mining of Both Positive and Negative Association Rules, ACM Transactions on Information Systems, Vol. 22, No. 3, July 2004, Pages 381–405.
[41] Zaki, M. J., Parallel and distributed association mining: A survey. IEEE Concurrency, Special Issue on Parallel Mechanisms for Data Mining, 7(4):14--25, December 1999.

Biography

S. Kotsiantis received a diploma in mathematics, a Master and a Ph.D. degree in computer science from the University of Patras, Greece. His research interests are in the field of data mining and machine learning. He has more than 50 publications to his credit in international journals and conferences.	**D. Kanellopoulos** received a diploma in electrical engineering and a Ph.D. degree in electrical and computer engineering from the University of Patras, Greece. He has more than 40 publications to his credit in international journals and conferences.