International Journal of Enterprise Computing and Business Systems
ISSN (Online) : 2230-8849
Volume 4 Issue 2 July 2014
International Manuscript ID : ISSN22308849-V4I2M1-072014

AN EFFECTIVE ALGORITHMIC APPROACH FOR CLASSIFICATION AND RULE MINING USING ONTOLOGY

Anuradha Rehal, M.Tech. Scholar,
Department of Computer Science and Applications
Kurukshetra University, Kurukshetra

ABSTRACT: *In Data Mining, the effectiveness of association rules is strappingly limited by the huge quantity of delivered rules. In this manuscript, we propose a new approach to prune and filter discovered rules. Using this approach of domain ontologies, the manuscript enriches the integration of user knowledge in the post-processing task. Moreover, an interactive and iterative framework is designed to assist the user along the analyzing task. On the one hand, we propose the user domain knowledge using Domain Ontology over database. On the other hand, a novel technique is suggested to prune and to filter discovered rules. In this manuscript, an effective methodology is proposed to evaluate and approximate the ontologies and their effectiveness in the real world scenario. The manuscript focus on medical data test set for detailed analysis and formation of the ontologies. In this research work, we have implemented and associated a metaheuristic ant colony optimization for efficient results in classification.*

Keywords *– Data Mining, Ontology, Association Rule Mining, Associative Classification, Ant colony Optimization*

I. INTRODUCTION

Association rule mining and standards, initially presented in 1993, are utilized to distinguish connections around a set of things in a database. These connections are not dependent upon innate properties of the information themselves (as with practical conditions), but instead dependent upon co-event of the information things. Association principles are additionally utilized for different requisitions, for example, expectation of disappointment in telecommunications arranges by distinguishing what occasions happen before a disappointment. The vast majority of our attention in this paper will be on bushel market dissection, however in later segments we will take a gander at different provisions too. Acquaintanceship guidelines are a standout amongst the most investigated zones of information mining and have as of late accepted much consideration from the database group. They have ended up being very convenient in the promoting and retail groups and other more various fields. In this paper we give a review of acquaintanceship guideline research.

The technique of mining association rules, introduced in [1], is considered as one of the most relevant tasks in Knowledge Discovery in Databases [2]. It aims to discover, among sets of items in transaction databases, implicative tendencies that can be revealed as being valuable information.

International Journal of Enterprise Computing and Business Systems
ISSN (Online) : 2230-8849
Volume 4 Issue 2 July 2014
International Manuscript ID : ISSN22308849-V4I2M1-072014

An association rule is described as the implication X → Y where X and Y are sets of items and $X \cap Y = \phi$. The strength of association rule mining rests in its ability to deliver interesting discovered knowledge that exists in data. Unfortunately, due to high dimensionality of massive data, this strength becomes its main weakness when analyzing the mining result. The huge number of discovered rules makes very difficult for a decision maker to manually outline the interesting rules. Thus, it is crucial to help the decision maker with an efficient reduction of the number of rules.

It is one of the major and critical area of data mining.

Data mining (the analysis step of the "Knowledge Discovery in Databases" process, or KDD),[1] an interdisciplinary subfield of computer science, [3][4][5] is the computational process of discovering patterns in large data sets involving methods at the intersection of artificial intelligence, machine learning, statistics, and database systems.[3]

To overcome this drawback, the post-processing task was proposed to improve the selection of discovered rules. Different complementary post-processing methods may be used, like pruning, summarizing, grouping or visualization. The pruning phase consists of removing uninteresting or redundant rules. In the summarizing phase summaries of rules are generated. Groups of rules are produces in the grouping phase; meanwhile the visualization phase is useful to have a better presentation.

However, most of existing post-processing methods are generally based on statistical information on database. Since rule interestingness strongly depends on user knowledge and goals these methods are not efficient enough. For instance, if the user looks for unexpected rules, all the already known rules should be pruned. Or, if the user wants to focus on specific schemas of rules, only this subset of rules should be selected.

In the terminology of machine learning [6] classification is considered an instance of supervised learning, i.e. learning where a training set of correctly identified observations is available.

Association rule learning is a popular and well researched method for discovering interesting relations between variables in large databases. It is intended to identify strong rules discovered in databases using different measures of interestingness. [9]

International Journal of Enterprise Computing and Business Systems
ISSN (Online) : 2230-8849
Volume 4 Issue 2 July 2014
International Manuscript ID : ISSN22308849-V4I2M1-072014

II. CRUCIAL ALGORITHMS

Most algorithms used to identify large item sets can be classified as either sequential or parallel. In most cases, it is assumed that the item sets are identified and stored in lexicographic order (based on item name). This ordering provides a logical manner in which item sets can be generated and counted. This is the normal approach with sequential algorithms. On the other hand, parallel algorithms focus on how to parallelize the task of finding large item sets.

SEQUENTIAL ALGORITHMS

AIS - The AIS algorithm was the first published algorithm developed to generate all large item sets in a transaction database. Artificial Immune Systems (AIS) are adaptive systems, inspired by theoretical immunology and observed immune functions, principles and models, which are applied to problem solving.[12] It focused on the enhancement of databases with necessary functionality to process decision support queries. This algorithm was targeted to discover qualitative rules. This technique is limited to only one item in the consequent. That is, the association rules are in the form of $X \Rightarrow Ij \mid \alpha$, where X is a set of items and Ij is a single item in the domain I, and α is the confidence of the rule. The AIS algorithm makes multiple passes over the entire database. During each pass, it scans all transactions.

SETM - The SETM algorithm [13] was proposed in and was motivated by the desire to use SQL to calculate large item sets [Srikant1996b]. In this algorithm each member of the set large item sets, $\overline{L_k}$, is in the form <TID, itemset> where TID is the unique identifier of a transaction. Similarly, each member of the set of candidate item sets, $\overline{C_k}$, 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 item sets by extending large item sets of the previous pass.

Apriori - The Apriori algorithm [13] developed by is a great achievement in the history of mining association rules. It is by far the most well-known association rule algorithm. This technique uses the property that any subset of a large itemset must be a large itemset. Also, it is assumed that items within an itemset are kept in lexicographic order. The fundamental differences of this algorithm from the AIS and SETM algorithms are the way of generating candidate item sets and the selection of candidate item sets for counting. As mentioned earlier, in both the AIS and SETM algorithms, the common item sets between large item sets of the previous pass and items of a transaction are obtained. These common item sets are extended with other individual items in the transaction to generate candidate item sets. However, those individual items may not be large. As we know that a superset of one large itemset and a small itemset will result in a small itemset, these techniques generate too many candidate item sets which turn out to be small. The Apriori algorithm addresses this important issue. The Apriori generates the candidate item sets by joining the large item sets of the previous pass and deleting those subsets which are small in the previous pass without considering the transactions in the

database. By only considering large item sets of the previous pass, the number of candidate large item sets is significantly reduced.

Apriori-TID - As mentioned earlier, Apriori scans the entire database in each pass to count support. Scanning of the entire database may not be needed in all passes. Based on this conjecture, proposed another algorithm called Apriori-TID [15]. Similar to Apriori, Apriori-TID uses the Apriori's candidate generating function to determine candidate item sets before the beginning of a pass. The main difference from Apriori is that it does not use the database for counting support after the first pass. Rather, it uses an encoding of the candidate item sets used in the previous pass denoted by $\overline{C_k}$. As with SETM, each member of the set $\overline{C_k}$ is of the form <TID, Xk> where Xk is a potentially large k-itemset present in the transaction with the identifier TID. In the first pass, $\overline{C_1}$ corresponds to the database. However, each item is replaced by the itemset. In other passes, the member of $\overline{C_k}$ corresponding to transaction T is <TID, c> where c is a candidate belonging to Ck contained in T. Therefore, the size of $\overline{C_k}$ may be smaller than the number of transactions in the database. Furthermore, each entry in $\overline{C_k}$ may be smaller than the corresponding transaction for larger k values. This is because very few candidates may be contained in the transaction. It should be mentioned that each entry in $\overline{C_k}$ may be larger than the corresponding transaction for smaller k values [Srikant1996b].

Off-line Candidate Determination (OCD) - The Off-line Candidate Determination (OCD) [16] technique is proposed in based on the idea that small samples are usually quite good for finding large item sets. The OCD technique uses the results of the combinatorial analysis of the information obtained from previous passes to eliminate unnecessary candidate sets. To know if a subset $Y \subseteq I$ is infrequent, at least (1-s) of the transactions must be scanned where s is the support threshold. Therefore, for small values of s, almost the entire relation has to be read. It is obvious that if the database is very large, it is important to make as few passes over the data as possible. OCD follows a different approach from AIS to determine candidate sets. OCD uses all available information from previous passes to prune candidate sets between the passes by keeping the pass as simple as possible.

CARMA - CARMA (Continuous Association Rule Mining Algorithm) [17] brings the computation of large item sets online. Being online, CARMA shows the current association rules to the user and allows the user to change the parameters, minimum support and minimum confidence, at any transaction during the first scan of the database. It needs at most 2 database scans. Similar to DIC, CARMA generates the item sets in the first scan and finishes counting all the item sets in the second scan. Different from DIC, CARMA generates the item sets on the fly from the transactions. After reading each transaction, it first increments the counts of the item sets which are subsets of the transaction. Then it generates new item sets from the transaction, if all immediate subsets of the item sets are currently potentially large with respect to the current minimum support and the part of the database that is read. For more accurate prediction of whether an itemset is potentially large, it calculates

International Journal of Enterprise Computing and Business Systems
ISSN (Online) : 2230-8849
Volume 4 Issue 2 July 2014
International Manuscript ID : ISSN22308849-V4I2M1-072014

an upper bound for the count of the itemset, which is the sum of its current count and an estimate of the number of occurrences before the itemset is generated. The estimate of the number of occurrences (called maximum misses) is computed when the itemset is first generated.

Table 1 - Comparison of Algorithms on multiple parameters

Algorithm	**Scan**	**Comments**
AIS	m+1	Suitable for low cardinality sparse transaction database; Single consequent
SETM	m+1	SQL compatible
Apriori	m+1	Transaction database with moderate cardinality; Outperforms both AIS and SETM; Base algorithm for parallel algorithms
Apriori-TID	m+1	Very slow with larger number of $\overline{C_k}$; Outperforms Apriori with smaller number of $\overline{C_k}$;
Apriori-Hybrid	m+1	Better than Apriori. However, switching from Apriori to Apriori-TID is expensive; Very crucial to figure out the transition point.
OCD	2	Applicable in large DB with lower support threshold.
Partition	2	Suitable for large DB with high cardinality of data; Favors homogenous data distribution
Sampling	2	Applicable in very large DB with lower support.
DIC	Depends on interval size	Database viewed as intervals of transactions; Candidates of increased size are generated at the end of an interval
CARMA	2	Applicable where transaction sequences are read from a Network; Online, users get continuous feedback and change support and/or confidence any time during process.
CD	m+1	Data Parallelism.
PDM	m+1	Data Parallelism; with early candidate pruning
DMA	m+1	Data Parallelism; with candidate pruning

Table 2 - Comparison of Algorithms on multiple parameters

Algorithm	**Scan**	**Data structure**	**Comments**
CCPD	m+1	Hash table and tree	Data Parallelism; on shared-memory machine
DD	m+1	Hash table and tree	Task Parallelism; round- robin partition
IDD	m+1	Hash table and tree	Task Parallelism; partition by the first items
HPA	m+1	Hash table and tree	Task Parallelism; partition by hash function

SH	m+1	Hash table and tree	Data Parallelism; candidates generated independently by each processor.
HD	m+1	Hash table and tree	Hybrid data and task parallelism; grid parallel architecture

III. ONTOLOGIES AND DATA MINING

Ontologies [7], introduced in data mining for the first time in early 2000, can be used in several ways [14]: Domain and Background Knowledge Ontologies, Ontologies for Data Mining Process, or Metadata Ontologies. Background Knowledge Ontologies organize domain knowledge and play important roles at several levels of knowledge discovery process. Ontologies [8] for Data Mining Process codify mining process description and choose the most appropriate task according to the given problem; meanwhile, Metadata Ontologies describe the construction process of items. Related to Generalized Association Rules, the notion of raising was exposed. Raising is the operation of generalizing rules (making rules more abstract) in order to increase support in keeping confidence high enough. This allows for strong rules to be discovered and also to obtain sufficient support for rules that, before raising, would not have minimum support due to the particular items they referred to. The difference with Generalized Association Rules is that this solution proposes to use a specific level for raising and mining.

The domain of data mining (DM) deals with analyzing different types of data. [18] A large-scale representation of abstract concepts such as actions, time, physical objects and beliefs would be an example of ontological engineering. [19]

Ontological engineering is a new field of study concerning the ontology development process, the ontology life cycle, the methods and methodologies for building ontologies [20] [21]

Ontology engineering aims at making explicit the knowledge contained within software applications, and within enterprises and business procedures for a particular domain. Ontology engineering offers a direction towards solving the inter-operability problems brought about by semantic obstacles, i.e. the obstacles related to the definitions of business terms and software classes. Ontology engineering is a set of tasks related to the development of ontologies for a particular domain - Line Pouchard, Nenad Ivezic and Craig Schlenoff, Ontology Engineering for Distributed Collaboration in Manufacturing [22]

IV. THE PROPOSED FRAMEWORK (NOVEL ALGORITHMIC APPROACH)

In the proposed methodology, as the case study, the information set of medical therapeutic records is entered in the back – end database. The improvement of co-operations and the cosmology is completely progressive. In the event that, any indication is looked, the proposed methodology looks from the back end database and makes the metaphysics that is dynamic in execution. It alludes that the unsupervised methodology of philosophy era is executed so the unprejudiced effects might be accomplished.

International Journal of Enterprise Computing and Business Systems
ISSN (Online) : 2230-8849
Volume 4 Issue 2 July 2014
International Manuscript ID : ISSN22308849-V4I2M1-072014

The proposed work is based on a well known metaheuristic ant colony optimization [10] for efficient results. This algorithm is a member of the ant colony algorithms family, in swarm intelligence methods, and it constitutes some metaheuristic optimizations. Initially proposed by Marco Dorigo in 1992 in his PhD thesis,[1][2] the first algorithm was aiming to search for an optimal path in a graph, based on the behavior of ants seeking a path between their colony and a source of food. [10] [11]

Procedure Complexity-Efficient-Classification (T, RSupport) { //T is the database and RSupport is the support

Activate Ai (Ant) Objects)
Initialize the Pheromone (Phi)
for 1 : n (Max. Ants and Pheromone Validity)
execute
F1= {frequent data items};
RIs={Required Item Set Support}
for (k = 3; fk-1 != RIs; k++) {
Ck= candidates generated from Lk-1
//that is Cartesian product Fk-1 x Kk-1 and eliminating any k-1 size itemset that is not
//frequent
for each transaction t in database do{
#increment the count of all candidates in Ck that are contained in t
Fk = candidates in Ck with RSupport
}//end for each
}//end for
Analyze the Pheromone Level and Status of Ants
End For
return Uk Fk ;
Analyze the Final Status of Pheromone Level and Ants
}
Procedure Time-Efficient-Classification(T, minSupport) {//T is the database and minSupport is the minimum support
Activate Ai (Ant) Objects)
Initialize the Pheromone (Phi)
for 1 : n (Max. Ants and Pheromone Validity)
execute
F1= {frequent items};
for (k= 2; Lk-1 !=ϕ; k++) { /*meaning of Lk and ϕ*/

International Journal of Enterprise Computing and Business Systems
ISSN (Online) : 2230-8849
Volume 4 Issue 2 July 2014
International Manuscript ID : ISSN22308849-V4I2M1-072014

Ck= candidates generated from Lk-1
//that is Cartesian product Lk-1 x Lk-1 and eliminating any k-1 size itemset that is not frequent
for each transaction t in database do{
#increment the count of all candidates in Ck that are contained in t
Lk = candidates in Ck with minSupport
}//end for each
}//end for
Analyze the Final Status of Pheromone Level and Ants
return Uk Lk ;
}

The new approach defines a new formal environment to prune and group discovered associations integrating knowledge into specific mining process of association rules. Firstly, a basic mining process is applied over data extracting a set of association rules. Secondly, the knowledge base allows formalizing user knowledge and goals. Domain knowledge allows a general view over user knowledge in database domain, and user expectations express user already knowledge over the discovered rules. Finally, the post-processing step consists in applying several operators (i.e. pruning) over user expectations in order to extract the interesting rules.

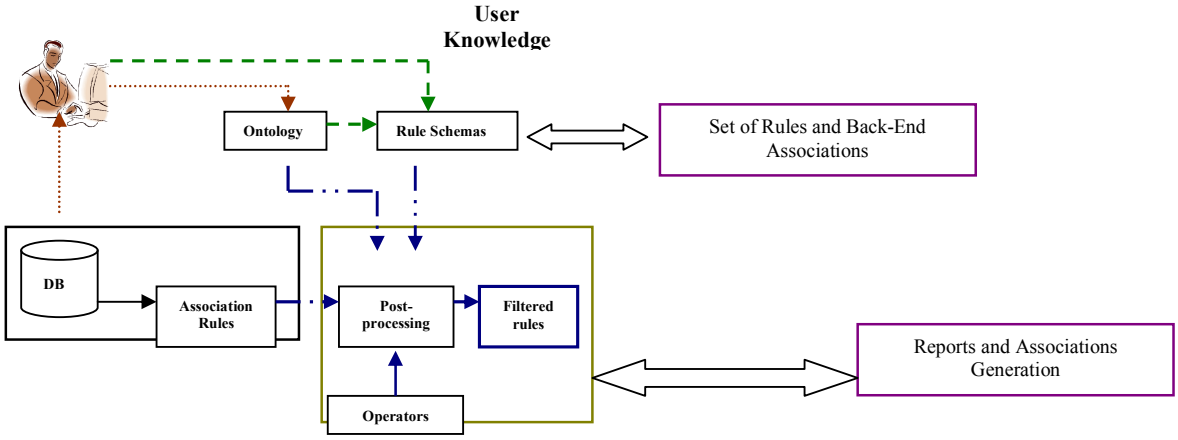

Figure 1 - Framework Description

The novelty of this approach resides in supervising the knowledge discovery process using different conceptual structures for user knowledge representation: one or several ontologies and several rule schemas.

V. DATABASE AND ASSOCIATION RULE MINING

The association rules mining techniques are applied over databases described as D = {I, T}. Let I = { I1, I2, ... ,Ip } be the set of attributes (called items) and T = { t1, t2, … tn } be the transaction set. Each transaction ti = { I1, I2, ... ,Imi } is a set of items, such as $t_i \subset I$ and each subset of items, X, is called itemset.

An association rule is an implication $X \rightarrow Y$, where X and Y are two item sets and $X \cap Y = \phi$. This rule holds on D with the confidence c if c% of transactions in T that contain X, also contain Y. The rule has support s in transaction set T if s% of transactions contain $X \cup Y$.

Since their early definition, association rules are mined using Apriori algorithm proposed for the first time in Agrawal et al., 1993.

VI. OPERATIONS IN POST-PROCESSING STEP

The post-processing task that we design is based on operators applied over rule schemas allowing to user to perform several actions over the discovered rules. We propose two important operators: pruning and filtering association rules. The filtering operator is composed by three operators: conforming, unexpectedness and exception.

VIII. CONCLUSION AND FUTURE WORK

This paper propose an effective methodology for the ontology design and discusses the problem of helping the decision maker in the post-processing step of association rule mining. The proposed work can be implemented in any domain including medical database, market basket analysis, web server log files and many others. The manuscript proposes to prune and filter discovered rule integrating user knowledge and beliefs. User knowledge is modelled in an ontology connected to data. Rule schemas allow user belief representation, and, combined with ontologies, they improve the selection of interesting rules. The manuscript intends to improve this approach in two directions - Developing the rule schema formalism and integrating the approach in the discovery algorithm. A number of techniques are used for optimization of the results related to classification and rule mining. The approach can be enhanced using high performance parallel algorithms and computing paradigms. The future scope of work includes execution of the algorithmic approach on hybrid approaches making use of grids and parallel systems.

IX. REFERENCES

[1] Agrawal, R., T. Imielinski, and A. Swami. (1993). Mining Association Rules between Sets of Items in Large Databases. Proceedings of the 12th ACM SIGMOD International Conference on Management of Data, pages 207 - 216.

[2] Fayyad, U. M., G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy. (1996) Advances in Knowledge Discovery and Data Mining, AAAI/MIT Press.

[3] Data Mining Curriculum". ACM SIGKDD. 2006-04-30.

[4] Clifton, Christopher (2010). "Encyclopædia Britannica: Definition of Data Mining". Retrieved 2010-12-09.

[5] Hastie, Trevor; Tibshirani, Robert; Friedman, Jerome (2009). "The Elements of Statistical Learning: Data Mining, Inference, and Prediction". Retrieved 2012-08-07.

[6] Anderson,T.W. (1958) An Introduction to Multivariate Statistical Analysis, Wiley.

[7] Gruber, Thomas R. (June 1993). "A translation approach to portable ontology specifications" (PDF). Knowledge Acquisition 5 (2): 199–220. doi:10.1006/knac.1993.1008.

[8] Arvidsson, F.; Flycht-Eriksson, A. "Ontologies I".

[9] 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.

[10] A. Colorni, M. Dorigo et V. Maniezzo, Distributed Optimization by Ant Colonies, actes de la première conférence européenne sur la vie artificielle, Paris, France, Elsevier Publishing, 134-142, 1991.

[11] M. Dorigo, Optimization, Learning and Natural Algorithms, PhD thesis, Politecnico di Milano, Italy, 1992.

[12] de Castro, Leandro N.; Timmis, Jonathan (2002). Artificial Immune Systems: A New Computational Intelligence Approach. Springer. pp. 57–58. ISBN 978-1-85233-594-6.

[13] Comparison and Analysis of Algorithms for Association Rules, IEEE Database Technology and Applications, 2009 First International Workshop, Comput. Sci. Dept., Dalian Nat. Univ., Dalian, China.

[14] Rakesh Agrawal and Ramakrishnan Srikant Fast algorithms for mining association rules in large databases. Proceedings of the 20th International Conference on Very Large Data Bases, VLDB, pages 487-499, Santiago, Chile, September 1994.

[15] A high efficient AprioriTid algorithm for mining association rule, IEEE Sch. of Electron. Inf. Eng., Tianjin Univ., China, Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference.

[16] Background for Association Rules and Cost Estimate of Selected Mining Algorithms, Jia, Ashley, ACM 2013.

[17] C. Hidber Online Association Rule Mining. In Proceedings of the ACM SIGMOD International Conference on Manag ement of Data, 2014.

[18] The OntoDM ontology, URL : http://www.ontodm.com/doku.php 2013

[19] URL : http://ontology.buffalo.edu/bfo/BeyondConcepts.pdf

[20] Asunción Gómez-Pérez, Mariano Fernández-López, Oscar Corcho (2004). Ontological Engineering: With Examples from the Areas of Knowledge Management, E-commerce and the Semantic Web. Springer, 2004.

[21] Denicola, A; Missikoff, M; Navigli, R (2009). "A software engineering approach to ontology building". Information Systems 34 (2): 258. doi:10.1016/j.is.2008.07.002.

[22] Line Pouchard, Nenad Ivezic and Craig Schlenoff (2000) "Ontology Engineering for Distributed Collaboration in Manufacturing". In Proceedings of the AIS2000 conference, March 2000.