# Mining Association Rules using Domain Ontology and Hefting

A.Razia Sulthana
Department of Information Technology
SRM University
Kattankulathur, Chennai

R.Subburaj
Department of Information Technology
SRM University
Kattankulathur, Chennai

## ABSTRACT

One of the major concerns in the field of knowledge discovery is the interestingness problem and the unreasonable number of association rules being mined. The past studies confirm that although a large number of rules are mined for each query, they do not seem to satisfy user's expectations. The methods already proposed in the literature like post-processing and algorithms to reduce itemsets and nonredundant rules do not always guarantee mining of interesting rules for the user. In conventional Data Mining, the usefulness of association rules is limited by the huge amount of delivered rules. In this paper we propose a new interactive approach 'Onto-Mine' to trim and filter the discovered rules. We propose to integrate user knowledge in association rule mining by combining Domain Ontology and interactive intelligence. First, we use Domain and Background Ontology with user knowledge and this interactive intelligence of Onto-Mine assists the user throughout the analyzing task and helps the user in selection of rules and also to revise the information that is proposed. Moreover ranking algorithm is used for retrieval of frequently accessed rules and the concept of privacy is enforced while mining. By applying the proposed approach the number of rules will be considerably reduced improving user productivity.

**Keywords** — Association Rule, Colander, FP Tree, Hefting, item set, Onto-Mine, Knowledge Discovery, Post mining.

## 1. INTRODUCTION

**I**t is recognized in the knowledge discovery in Data (KDD) [1] research and applications that rules are the most expressive and human understandable representations of knowledge i.e. rules provide a self explanatory result to the user. The development of World Wide Web resulted in the internet being accessible to millions of people since it is easy for anyone to access information from it. The rapid growth of a number of documents is leading to serious problem of information overload. So, rule discovery is considered to be the most important issue in data mining and in machine learning.

Association rule [2] discovery process produces a comprehensive rule set with the rules satisfying the minimum threshold value. Association rule discovery is a general-purpose rule-discovery scheme and has many applications. The threshold measures of association rules are Support and Confidence.

An association rule is represented by an implication P→c where p is the antecedent and c is the consequence. An association rule is said to be strong if it satisfies both the minimum Support and Confidence. There are two phases in finding association rules. In the first phase we find the set of frequent itemsets and in the second phase we use the frequent itemsets to generate the interesting patterns. Apriori [3] is the one of the earliest algorithm to mine association rules where to make the rules more interesting for the user, the support threshold must be minimum. Minimum the support threshold, the maximum the number of rules mined. But, as the number of rules exceeds a count of say, 100 it creates a difficult situation for the user to find out the satisfactory rule.

In order to reduce the voluminous number of rules many approaches have been proposed. For instance, mining the rules using the deductive method interacts with the user frequently by making them to pick the interesting items. And few techniques make use of taxonomies for reducing only the hierarchical redundant rules in multilevel datasets. By generating closed, optimal, maximal [4] and frequent itemsets many algorithms try to reduce the number of rules. Postprocessing methods like pruning, summarizing, grouping and visualization are used in existing methods. The rules are expressed to the user in a more efficient, accurate and in a flexible manner so as to easily identify them.

The use of ontologies in semantic web enables rather quick and accurate web search. It also allows the development of intelligent Internet agents and facilitates communication between multitudes of heterogeneous web-accessible devices. The existing post processing methods depend on the analytical information, which does not prove that the mined rules are interesting to the user.

In this paper the authors propose a new approach called "Onto-Mine" to trim the discovered rules. Onto-Mine combines Domain ontologies and interactive intelligence. We propose domain and background ontologies for binding the user knowledge during the post processing [5] step. Interactive intelligence guides the user throughout the process using iterative loop in analyzing the task. Also the ranking algorithm will provide the relationship amongst concepts embedded into semantic annotations. This sort of ranking functions[6] at an inner level (i.e., it exploits more precise information that can be made available within a Web page).

Our approach only relies on the knowledge of the user query, the Web pages to be ranked, and the underlying ontology. Thus, it allows the miner to effectively manage the search space and to reduce the complexity associated with the ranking task. Privacy is provided while retrieving the data from the database. The exposure of the inner data depends only on the rights of the user. Only an authentic person can access the secure data.

The paper is organized as follows. The related research work is given in section 2. Section 3 describes the proposed framework-Onto-Mine. In Section 4 and section 5 we discuss the results obtained using Onto-Mine and the conclusions arrived at respectively.

# 2. RELATED WORK

The ORD (Optimal Rule Discovery) algorithm [7] proposes to mine the optimal rule sets. It generates a unified framework for the discovery of a family of optimal rule sets and the relationships with other rule-discovery schemes such as non redundant association rule. Moreover, it deals only with the heuristic rule discovery and not with association rule discovery.

Deductive method [8] of mining rules mines frequent itemset starting from candidate two-itemsets to candidate (n-1) itemsets with inductive method and produces huge rough rules on these frequent item sets. It reduces producing huge amount of frequent itemsets. Moreover, it needs dynamic interaction with the user whenever the user wants to check whether their interesting patterns were selected.

User-Expectation method [9] finds interesting patterns and also reduces the number of rules mined. In this technique, the user is first asked to provide his/her expected patterns according to his/her past knowledge or intuitive feelings. Given these expectations, the system uses a fuzzy matching technique to match the discovered patterns against the user's expectations, and then rank the discovered patterns according to the matching results. A variety of rankings are performed for different purposes, such as to confirm the user's knowledge and to identify unexpected patterns, which are by definition interesting. This method does not solve the problem associated with unexpected measures.

The MAFIA [4] algorithm is based on depth-first-traversal and several pruning methods. It combines a vertical bitmap representation of the data with an efficient bitmap compression scheme. MAFIA also generates all frequent itemsets and closed frequent itemsets, though the algorithm is optimized for mining only maximal frequent itemsets. It uses many pruning techniques PEP (Parent Equivalence Pruning), FHUT, HUTMFI or dynamic reordering. The drawback with MAFIA is the loss of information because the subset frequency is not available; also it requires more space in memory to fit the entire database completely in memory.

Another measure mines the association rule based upon any-confidence, all-confidence and bond [10]. The downward closure property is applied to both all-confidence and bond. It also proves that associations that have a minimum all-confidence or minimum bond will have a 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. It uses association finding algorithm to generate large itemsets. The drawback of this method is that it fails to find itemsets which are small.

ML_T2L1 [11] algorithm is an adaptation of Apriori to multi-level datasets. Association rule mining plays an important role. For multi-level datasets the number of discovered rules is large and huge. This approach eliminates the redundant rules from multilevel datasets. This approach modifies the Apriori algorithm to work with multi-level datasets which is designed for single-level datasets. But the ML_T2L1 does not find cross level itemsets.

## 2.1 User Driven Mining

Interestingness Measures [12] discover the rules that are interesting to the user. Subjective and Objective measures are the two different types. Objective Measures that depend only on the data structure of a pattern and the underlying data used in the discovery process. Subjective Measures depend only on the class of the user who examine the pattern. But the objective measures are not sufficient to reduce the number of extracted rules and to capture the interesting ones. Unexpectedness and actionability are the types of subjective measures proposed by Silbershatz and Tuzlin [13]. Unexpectedness-a pattern is interesting if it is surprising to the user. Actionability-a pattern is interesting if the user can act on it to his advantage.

Klemttinen [14] proposes templates to describe interesting rules and non-interesting rules. Other approach uses rule like formalism to express user expectation and mines the rules on comparing to the user expectation.

## 2.2. Ontologies in Mining

Ontology [15] is a widely adopted key technology for intelligent knowledge processing, providing a concept system of certain domain, which can reuse prior knowledge and reduce or eliminate confusion of concepts or terms. The ontology is introduced by German Philosophers in Greek where ontos means “being” and logos means “word”.

In the early 1990s, ontology was defined by Gruber as a formal, explicit specification of a shared conceptualization [15]. By conceptualization we understand an abstract model, explicit means that the elements are clearly defined and formal means that the ontology should be machine processable.

Ontology is a “formal specification for the intended mining of a formal vocabulary”.

There are several ways of using Ontologies [16]: 1. Domain and Background Ontologies 2. Ontologies of data mining process 3. Metadata Ontologies. In this paper, we focus on the Domain and Background ontologies introduced by Srikant and Agrawal [17]. The advantage of the Domain and Background Ontologies is that it can benefit all phases of KDD cycle.

Hefting is the operation that makes rules more abstract and in keeping confidence high enough it increases support. Also hefting discovers strong rules which possess low support before.

There are four types of Ontologies proposed in literature [18]: 1. Upper (or top level) Ontologies 2. Domain Ontologies 3. Task Ontologies 4. Application Ontologies. The upper ontology deals with the general concept and the other three types deal with domain specific concepts.

A few ontology repositories [19] are developed for information system including ontoserver, webonto and ontolingua. They provide a repository of ontologies to assist users in generating new ontologies and in managing the existing ontologies.

# 3. FRAMEWORK DESCRIPTION

In this paper we propose using the postprocessing method. Basically pruning and abstraction are the two steps done during the pre-processing. In the preprocessing task the process of applying the constraints is done in the first step. Mining is done in the second step. As constraints are applied in the first step, pruning excludes the rules that are interesting to the user . But in post processing we refine the queries based on syntax and semantics, so no rules are lost during query formulation.

The query of the specified user is considered in conjunction with the history of queries of not only the specified user but also other users which are stored in the query history database. The query is thus checked for syntax and then semantics. Thereafter the refined query is formulated.

Using the ontology's assistance the system will ensure for the correctness and effectiveness of the query been formulated. (Figure 1).

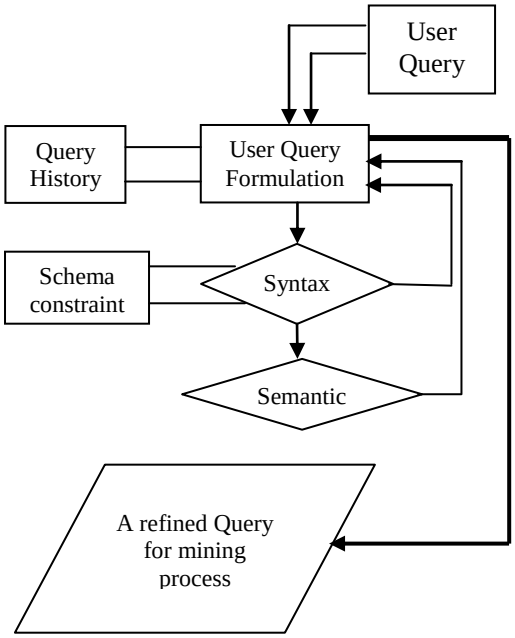


**Figure 1: Process of user query checking**

The proposed approach is composed of integration of Domain ontology and interactive intelligence as shown in Figure 2.

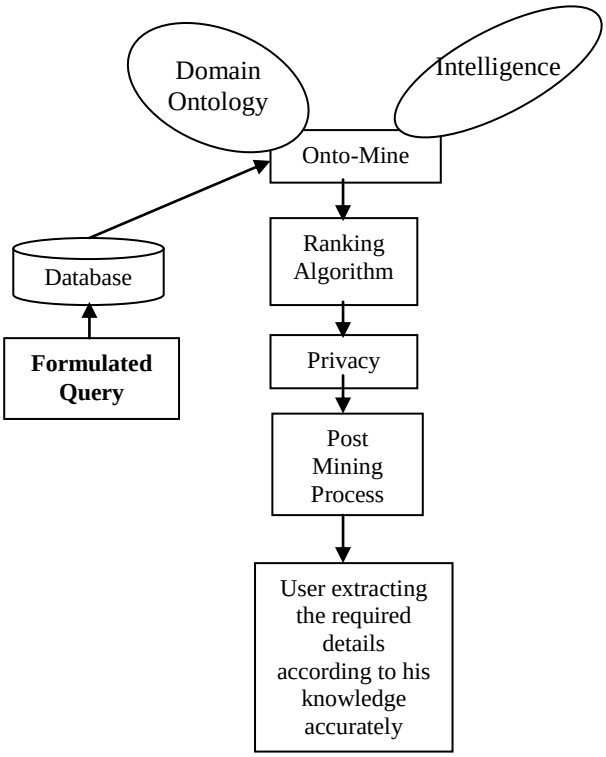


**Figure 2: Onto-Mine Framework Description**

The formulated query is sent to the database where Onto -Mine framework is applied which combines Domain Ontology and Interactive Intelligence. we find that the postprocessing step follows the ontology and this method applies iteratively a set of colander over the rules. There are two types of colander: Least progress Coercion colander, Item-akin colander. In this paper we use Item-Akin Colander and measure the idea of relatedness between the items and their similarities. Relatedness between the condition of the rule and the consequent of the rule is also measured.

There are three types of rule schema: General Impression (GI) [12], Reasonable Precise Concepts (RPC) and Precise Knowledge (PK). Onto-Mine makes use of GI rule schema. The syntax of GI is given by

GI (<i1, i2,….im>) [support, confidence]

where $s_i$ is an element of item taxonomy.

The main difference between the rule schemas GI and RPC is based on the implication characteristics. In GI the direction of the implication is not known, so it is not possible to find out the antecedent and consequent. But the RPC supports complete implication, so the consequent and the antecedent can be easily found. The problem with Precise Knowledge is that it is analytically defined and needs the exact support and confidence for rule schema because of which is not used in many cases for filtering. The concept of ontology is different from that of taxonomy. Taxonomy is a classification or hierarchical categorization of items in a domain. But ontology specifies many characteristics of a domain.

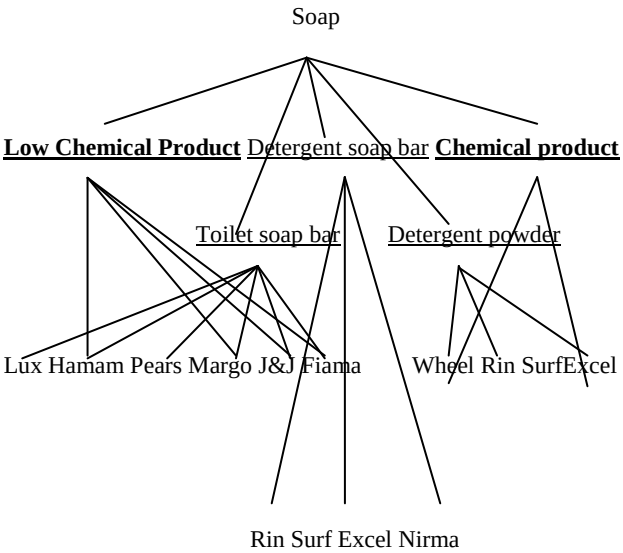


**Figure 3: Ontology Description**

The ontology description is given in Figure 3. The expressiveness for ontology is provided only by rule schemas by combining the abstraction and pruning constraints. The syntax of rule schema is given by

RS (<x1,….,xn (→) y1,…,ym>)

Where xi and xj are subset of C of O where O is the ontology which is {C, R, I, H, A}. And → is optional, which when used comes under implicative rule schema and when not used comes under non implicative rule schemas.

RS (Low Chemical Product, chemical Product>), which is a non implicative rule schema and comes under GI.

RS (<Low Chemical Product → chemical Product>), which is a implicative rule schema and comes under RPC.

The above rules are framed from the Figure 3. Also three concepts of ontology can be defined from them 1. Leaf concepts 2. Generalized concept 3. Restriction concepts. The Leaf concepts include the one in the leaf nodes of the figure. The generalized concept is defined as root at level 1 and level 2 and does not include the ontology concepts. Restriction concepts include only the ontology at level 2 in bold.

The paper makes use of four operators for filtering the rules. The types of operators that are used to filter these rules are 1. Trim 2. Correctness 3. Surprising 4. Exclusion. The representations of these operators are defined by T (RS), C (RS), S (RS), E (RS) respectively.

To filter the uninteresting rules the Trim operators are used. The correctness operator confirms the implication from the concepts in the database. It colanders the non-entailed rule schema. Rules which provide surprise effect to the user are filtered using the surprising filter. Confirming rules are mined from exclusion operators. The operators in rule schema are given in Figure 4.

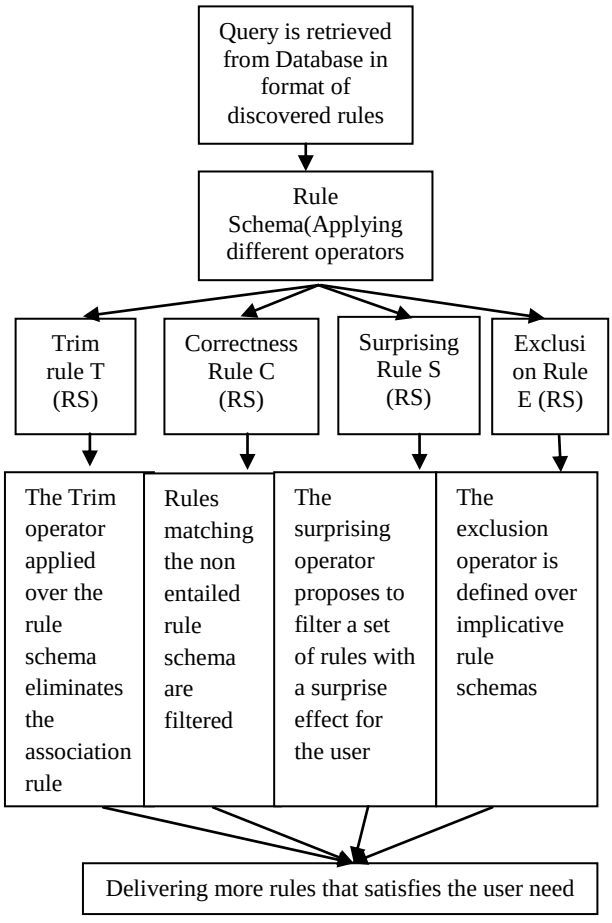


**Figure 4: Operators in Rule Schema**

Among the colander used LPCC (Least progress Coercion Colander) selects those rules whose confidence is greater than *minimp*. IAC (Item-Akin colander) measures the semantic distance between the item taxonomies.

In our approach we use the IAC filter because users are interested in finding association between items with different functionalities. We calculate the distance between the items by the minimum path that connects the two items. Thus the IAC calculates the minimum of all the distance computed among items in the condition and the consequent.

Finally the ranking algorithm defines the rank results where most of the solutions need to work on the whole annotated knowledge base. Relevance is measured as the probability that a retrieved resource actually contains those relations whose existence was assumed by the user at the time of querying. We propose a novel ranking strategy that provides a score (rank) for each web page.

On each access to the page simultaneously the rank for the page is incremented. Based on the user query the page with high rate is chosen and is considered as most frequently accessed web pages. Accordingly the rules with higher ranks are displayed on the top. Several ranking algorithm based on the relation based metadata have been proposed but they mainly use page relevance criteria based on the information from the knowledge base making the application infeasible in huge platforms. Relevance is measured as the probability that a retrieved resource actually contains those relations whose existence was assumed by the user at the time of querying.

The database contains the pages or concepts and the rules along with association rules been mined. Thus against each page the ranks are provided. It provides fast accessing technique by searching the database table. There is no necessity to scan the entire server and search for the number of times the particular rule is accessed. Also it contains information about the frequently accessed rules there by making the accessing fast.

The rules are aligned based on the rank for the user. Thus the rules having higher ranks are placed at the top and the rules at the bottom have low rank.

Also we implement the concept of privacy during mining the data from the database thereby the concept of security is implemented. This privacy concept does not expose all the data intended to the user but it exposes the data based upon the authorized data.

## 4. RESULTS

We tested Onto-Mine in a fruit dataset containing 1000 data on apple, 500 data on facts over fruits, 200 data on dates, 20 data on orange. When the proposed Onto-Mine was used with the above dataset and with the query as ‘Apple and its vitamins’, the results that were retrieved contained 100 rules as per details below.

1. 60% of association rules has apple’s vitamin information(using correctness operator),
2. 30% of association rules contains apple information and vitamin information separately(using trim operator),
3. 8% association rules has dates information as apple and dates are rich in iron (using surprising rule)
4. 2% association rules that contain facts about apple (using exclusion rule)

Thus on using proposed approach the number of rules filtered is less in number and are interesting to the user as we are using different operators to filter the interesting rules to satisfy the users need. In the conventional methodology the mining process filters to a minimum of 400 rules which is greater than the number of rules mined using our proposed method

## 5. CONCLUSION

This paper discusses mining interesting association rules from a huge voluminous number of rules. We propose Onto-Mine framework that incorporates Domain ontology, interactive intelligence, ranking approach and privacy in mining the rules. Also, the IAC colander filters out the unwanted rules using mining operators. These operators are used in the postprocessing task and guide the user through out the process. Onto-Mine framework prunes and filters the needed rules as per to the expectation of the user. It can be extended to mine rules using other filters. Thus, by using this approach we reduce the number of rules mined and ensure that the mined rules are interesting to the user.

## 6. REFERENCES

[1] Feyyad, U. M. 1996. Knowledge Discovery and Data Mining: Making Sense out of Data. Journal of IEEE Expert Magazine. Vol 11. Issue 5. Page 20-25.

[2] Jiawei Han. and Yongjian Fu. 1999. Mining Multiple Level Association Rules from Large Databases. IEEE Transactions on Knowledge and Data Engineering. Vol 11. Issue 5. Page 798-805.

[3] Argawal, R. and Imielinski, T. 1993. Mining Association Rules between Sets of Items in Large Databases. Proc ACMSIGMOD.

[4] Burdick, D. Calimlim, M. Flannick, J. Gehrke, J. and Yiu,T. 2005. Mafia: A Maximal Frequent Itemset Algotithm. IEEE Transactions on Knowledge and Data Engineering. Vol 17. No 11. Page 1490-1504.

[5] Qiang Yang. Jie Yin. Ling, C,X. Chen, T. 2003. Post Processing Decision Trees to Extract Actionable Knowledge. Third IEEE International Conference on Data Mining.

[6] Fabrizio Lamberti. Andrea Sanna and Claudio Demartini. 2009. A relation- based page rank algorithm for semantic web search engines. Vol 21. No 1. Page 123-136

[7] J. Li. On Optimal Rule Discovery. 2006. IEEE Transactions on Knowledge and Data Engineering. vol 18. no 4. Page 460-471.

[8] Chien-Le Goh. Tsukamoto, M. Nishio, S. 1996. Knowledge discovery in deductive databases with large deduction results: the first step. IEEE Transactions on Knowledge and Data Engineering. Vol 8. Issue 6. Page 952-956

[9] Bing Liu. Wynne Hsu. Lai-Fun Mun. and Hing_Yan Lee. 1999. Finding Interesting Patterns Using User Expectations. IEEE Transactions on Knowledge and Data Engineering. Vol 11. No 6. Page 811-831.

[10] E.R. Omiecinski. Alternative Interest Measures for Mining Associations in Databases. IEEE Trans. Knowledge and Data Engineering. Vol 15. No 1. Page 57-69.

[11] Zhang Hong. Zhang Bo. Kong Ling-Dong. Cai Zheng Xing. 2001. Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing.

[12] Jen-Wei Huang. Chi-Yao Tseng. Jian-Chih Ou. Ming-Syan Chen. 2008. A General Model for Sequential Pattern Mining with a Progressive Database. IEEE Transactions on Knowledge and Data Engineering. Vol 20. Issue 9. Page 1153-1167.

[13] Silberschatz, A. Tuzhilin, A. 1996. What Makes Patterns Interesting in Knowledge Discovery Systems. IEEE Transactions on Knowledge and Data Engineering. Vol 8. No 6. Page 970-974.

[14] M. Klemettinen, H. Mannila, P. Ronkainen, H. Toivonen. and A,I,Verkamo. 1994. Finding Interesting Rules from Large Sets of Discovered Association Rules. Proc. Int'l Conf. Information and Knowledge Management (CIKM). Page 401-407.

[15] Karin Koogan Breitman. 2003. Ontology as a Requirements Engineering Product. IEEE International Requirements Engineering Conference. Page 309-319.

[16] H. Nigro. S,G, Cisaro. and D. Xodo. 2007. Data Mining with Ontologies: Implementations, Findings and Frameworks. Idea Group, Inc., 2007.

[17] R, Srikant. and R, Agrawal. 1995. Mining Generalized Association Rules. Proc. 21st Int'l Conf. Very Large Databases. Page 407-419.

[18]N, Guarino. 1998. Formal Ontology in Information Systems. Proc. First Int'l Conf. Formal Ontology in Information Systems. Page 3-15.

[19] Ding Pan. Yan Pan. 2006. Using ontology repository to support data mining. Proc of the 6th world congress on intelligent control and automation.

## APPENDIX

Formally, the association rule is: Let $R = \{r_1, r_2, \ldots r_m\}$ be a set of m distinct literals called items. T is a set of variable length transactions over R. Each transaction contains a set of items $x_1$, $x_2 \ldots x_k \subset R$. An association rule is an implication of the form $A \rightarrow B$, where A, $B \subset R$ and $A \cap B = \varnothing$. A is called the antecedent and B is called the consequent of the rule.

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. Itemset of some length k are referred to as k-itemsets. For an itemset $A \cup B$, if B is an m-itemset then B is called an m-extension of A. The threshold measures of an association rule are support and confidence.

**Definition 1:**

The support of an implication is $P \rightarrow c$ is the ratio of the number of records containing both P and c to the number of records in D, denoted by supp ($P \rightarrow c$).

$$\text{Support } (P \rightarrow c) = \frac{\#\text{tuples containing both P and c}}{\#\text{total\_no\_of\_tuples}}$$

Support $P \rightarrow c) = P (P \cup c)$

**Definition 2:**

The confidence of the implication $P \rightarrow c$ is defined to be ratio of supp ($P \rightarrow c$) to supp (P), represented by conf ($P \rightarrow c$). The confidence forms the conditional probability.

$$\text{Confidence } (P \rightarrow c) = \frac{\#\text{tuples containing both P and c}}{\#\text{tuples\_containing\_P}}$$

Confidence $(P \rightarrow c) = P (P \mid c)$

Starting from the database minsupp and minconf are the two thresholds been defined and any rule whose support and confidence exceeds this value is considered to be the proper rule. This process is basically done in two steps

1. Initially the itemsets that are more frequent are extracted. An item is called so if support (P) >=minsupp.
2. Then for each frequent itemset the set of rules with conf (P-c→c) >=minconf is generated.

**Definition 3:**

A Frequent Itemset is defined as an itemset X which satisfies the minimum support count. The number of transactions required of the itemset to satisfy minimum count is called minimum support count.

**Definition 4:**

An Optimal rule is a subset of a non redundant rule set. A rule set is optimal with respect to interestingness metric if it contains all tuples except those with no greater interestingness than one of its more general rules.

**Definition 5:**

An Ontology is a quintuple (5-tuple) consisting of the core elements of an ontology, i.e., concepts, relations, hierarchy, a function that relates concepts non-taxonomically and a set of axioms. The elements are defined as follows:

O: = {C, R, $H^c$, rel, $A^o$} consisting of:

Two disjoint sets, C (*Concepts*) and R(Relations)

A Concept hierarchy, $H^c$: $H^c$ is a directed related $H^c$ C*C which is called concept hierarchy or taxonomy. $H^c$ (C1,C2) means $C_1$ is a sub concept of $C_2$.

A function rel: R→ C*C that relates the concept non taxonomically.

A set of ontology Axioms $A^o$, expressed in appropriate logical language.

## AUTHOR'S PROFILE

**R.Subburaj** is currently Professor and Consultant in the Department of Information Technology, SRM University, Chennai area. He played a key role in the Department of Electronics and Information Technology/STQC Directorate of the Government of India for nearly three decades. He Headed Electronics Test & Development Centre, Chennai, Centre For Reliability, Chennai and Electronics Regional Test Laboratory (West), Mumbai. He was authorized by Carnegie Mellon University/Software Engineering Institute, USA to teach their official Intro' to Software CMM course. He was awarded IETE Lal C Verman Award for the year 2003 for his distinguished contributions in the field of "Standardization, Quality Control and Precision measurements". He was awarded M.Tech degree by Indian Institute of Technology (IIT), Delhi, India and awarded Ph.D. degree by University of Madras, Chennai, India. He was the Conference Chair of the second International Conference on Reliability and Safety Engineering – INCRESE 2006.

**A. Razia Sulthana** is Assistant Professor in the Department of Information Technology, SRM University, Chennai area. She is pursuing her research in the area of data mining. She was awarded M.E and B.Tech degrees by Anna University, Chennai. She is a university rank holder during her post graduation studies in the Anna University.