Chapter 2

Literature Survey

2.1 Introduction

In this thesis, the literature survey covers period from 1993 to 2012. In the literature, different researchers have categorized the association rule mining techniques based on different ground. The most satisfactory classification of data mining techniques is on the basis of the layout of the database under consideration. Different approaches have been proposed that uses horizontal layout of database, vertical layout of database or projected layout of database. Some researchers work on improving the efficiency of the mining process while others tried to dig out advanced, complicated and high-level knowledge from the database. Similarly, swarm intelligence techniques have been used in different fields for various tasks ranging from optimization to distribution of resources. The use of swarm
intelligence for data mining has become popular since last two decades. After that several innovations in the field of data mining using swarm intelligence has been carried out. This chapter ponders light on the available literature in both the fields viz. data mining and swarm intelligence and also presents a discussion of the successful applications of different swarm intelligence techniques in data mining.

2.2 Evolution of Data Mining Techniques

Data and information has been accepted as a valuable asset since long time. But the utilization of data and the tools for using that data has been changed a lot over time. During 1960’s database creation and network DBMS were more popular. In 1970’s, the relational database model and relational DBMS implementations came into use. Advanced database models like deductive databases, extended-relational, object-oriented and application oriented DBMS had come into light in 1980’s. After that the explosive growth of data and tools for handling data gave birth to advanced technologies like data mining, data warehousing, multimedia databases and web databases. Around 1990’s, data mining gained lots of popularity. Several advanced technologies of data mining came into light and made the field of research centered on various data mining tasks. Since 2000, the boom of data mining was escalating and today it has captured the attention of a large community of researchers and it has become the well-researched area. The mark-able research activities in the field of data mining are discussed in the next sections.

2.3 Association Rule Mining

Association rule mining is one of the important and well researched techniques of data mining to find important correlations among data items. Based on the layout of database,
different techniques have been developed for mining the data. The horizontal layout of database is used by Apriori series methods whereas vertical layout is the base of FP-growth and Eclat algorithms. Other approaches either enhance the efficiency of the existing approaches or deals with high level data mining concepts. These well-liked approaches are discussed below:

2.3.1 Techniques based on Horizontal Layout of Databases

The first algorithm to generate all frequent itemsets was proposed by Agrawal et al. [AGR1993] and named AIS (after the name of its proposers Agrawal, Imielinski and Swami). The algorithm generates all the possible itemsets at each level of traversal. Thus, it generates and stores frequent as well as infrequent itemsets in each pass. Generation of infrequent itemsets was undesirable and was a major drawback over its performance. Later on, AIS was improved upon and renamed as Apriori by Agrawal et al. The new algorithm uses a level-wise and breadth-first search for generating association rules. Apriori and Apriori Tid algorithms generate the candidate itemsets by using only the itemsets found large in the previous pass and without using the transactional database. Apriori uses the downward closure property of the itemset support to prune the itemset lattice- the property that all subsets of frequent itemsets must themselves be frequent.

A similar algorithm called Dynamic Itemset Counting (DIC) was proposed by Brin et al. in [BRI1997]. DIC partitions a database into several blocks marked by start points and repeatedly scans the database. Unlike Apriori, DIC can add new candidate itemsets at any start point, instead of just at the beginning of new database scan. At each start point, DIC estimates the support of all itemsets that are currently counted and add new itemsets to the set.
if all its subsets are estimated to be frequent.

Part et al. in [PAR1995] proposed the Dynamic Hashing and Pruning (DHP) algorithm. DHP can be derived from Apriori by introducing additional control. For this purpose, DHP makes use of an additional hash table that aims at limiting the generation of candidates as much as possible. DHP also progressively trims the database by discarding attributes in transactions or even by discarding entire transactions when they appear to be subsequently useless. Therefore, DHP includes two major features, the efficient generation of large itemsets and the effective reduction of transaction database size.

Vu et al. [THN2008] proposed a rule based prediction technique to predict the user featured location, but this method generates more candidate itemsets than required. As the database must be scanned multiple times, the algorithm was expensive in terms of run time and I/O load.

The Pincer-Search algorithm was proposed by Lin et al. [LIN1997] and can efficiently discover the maximum frequent set. The Pincer-Search combines both the bottom-up and top-down directions. The main search direction is still bottom-up but a restricted search is conducted in the top-down direction. This search is used only for maintaining and updating the new data structure designed for this study, namely, the maximum frequent candidate set. It is used to prune the candidates in the bottom-up search. Another very important feature of this algorithm is that it is not necessary to explicitly examine every frequent itemset. Therefore, it performs well even when some maximal frequent sets are long. This algorithm reduces the number of times the database is scanned and the number of candidates considered.
The parallel algorithms for the discovery of association rules using clustering techniques to approximate the set of potentially maximal frequent itemsets was first proposed by Zaki et al. [ZAK1997]. This algorithm uses two clustering schemes based on equivalence classes and maximal hyper-graph cliques and study two lattice traversal techniques based on bottom-up and hybrid search and also use vertical database layout to cluster related transactions together.

Ashoka Savasere et al. [SAV1995] proposed the partition algorithm for mining association rules that is different from the classical algorithms. It requires two database scans to mine frequent itemsets and also reduce I/O overhead. First the partition algorithm scans the database once to generate a set of all potential frequent itemsets and in the second scan, their global support is obtained. The algorithm partitions the database into small chunks which can be held in main memory. The partitions are considered once at a time and all large itemsets are generated for that partition. These large itemsets are further merged to create a set of all potential large itemsets.

Mohammed J. Zaki and Ching-Jui Hsiao [ZAK1999] presented a new approach, CHARM, which introduces some lattice theory concepts to improve the mining. CHARM is unique in that it simultaneously explores both the item set space and tid (transaction id) set space, unlike all previous association mining methods which only exploit the item set space. It avoids enumerating all possible subsets of a closed item set when enumerating the closed frequent sets, which rules out a pure bottom-up search. This property is important in mining dense domains with long frequent item sets, where bottom-up approaches are not practical.
Performance dramatically decreases in mining process of many association rule algorithms. This is due to the fact that a database is repeatedly scanned to compare each candidate item set with the whole database level by level in the process of mining association rules. Yuh-Jiuan Tsay and Jiunn-Yann Chiang [TSA2005] introduced CBAR algorithm. CBAR only requires a single scan of the transaction database, followed by comparison with the partial cluster tables. This not only prunes considerable amounts of data reducing the time needed to perform data scans and requiring less contrast, but also ensures the correctness of the mined results.

2.3.2 Techniques based on Vertical Layout of Databases

Most of the algorithms that mine frequent itemsets use a horizontal data layout, however, many researcher use a vertical layout. The main advantage of using a vertical layout database is that it is easy to compute the support of any k-itemset by simple intersection of tidlist of the lexicographically ordered first (k-1)-itemsets.

The Eclat algorithm proposed by Zaki et al. [ZAK1997] uses prefix based classes and bottom-up search and generates all frequent itemsets in a breadth-first search using the joining step from the Apriori property when no candidate items can be found. This is an equivalence class based algorithm. The Eclat algorithm is very efficient for large data sets but is less efficient for small data sets. Other algorithms in the series are MaxEclat, Clique and MaxClique. MaxEclat uses prefix-based classes and hybrid search. Clique uses Clique-based classes and bottom-up search whereas MaxClique uses clique-based classes and hybrid search. The best approach is Maxclique which out-performe Apriori and Partition by an order of magnitude and Eclat by a factor of 2 or more.
Zaki et al. [ZAK2003] presented a novel fast vertical mining algorithm using diffsets that only keeps track of the differences in the tids of a candidate pattern from its generating frequent patterns. The results of experiments shows that diffsets drastically cut down the size of memory required to store the intermediate results. This improves the performance of mining process significantly.

Kan Jin [KAN2010] proposed an algorithm LogEclat based on Eclat and uses special candidates to find frequent patterns from a continually updating database. The algorithm can find several k-itemsets in one time of scanning database and thus the time of establishing new database is reduced. Experimental evaluation of the algorithm proves that LogEclat can find frequent patterns correctly and performs better than Apriori and Eclat.

2.3.3 Techniques based on Projected Layout of Database

Han J. and Pei J. [HAN2000a] introduced FP-Tree algorithm which uses the projected layout database. FP –Tree is frequent pattern mining technique that generates frequent itemsets 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 1-length 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. The authors claim that the FP-tree scales much better than Apriori.

A TreeProjection algorithm is proposed by V. Prasad and R. Agarwal [PRA2000]. The general idea of the algorithm as reported by the authors is that it constructs a lexicographical tree that projects a large database into a set of reduced, item-based sub-databases based on
frequent patterns mined so far. An efficient algorithm for mining association rules was then presented. The efficiency of TreeProjection algorithm can be determined from two major facts. First, the transaction projection limits the support counting in a small space and second the lexicographic tree facilitates the management and counting of candidate sets.

Another approach based on FP-tree and co-occurrence frequent items (COFI) was proposed by V.K. Shrivastava et al. [SHR2010] to find frequent itemsets in multilevel concept hierarchy by using a non-recursive mining process.

To improve upon the cost of main memory requirement, Pei et al. proposed another algorithm called H-mine using array and tree based data structure. A distinct feature of this method is that it has very limited and precisely predictable space overhead and runs really fast in memory-based setting. It can be scaled up to very large databases by database partitioning and when the dataset becomes dense, FP-tree can be constructed dynamically as part of the mining process [PEI, 2001].

Sahaphong and Boonjing proposed a new algorithm [SAH2008] which constructs a pattern base using a new method that is different from the patterns base in the FP-growth and mines frequent itemsets using a new combination method without the recursive construction of a conditional FP-tree.

Supatra Sahaphong and Veera Boonjing [SAH2012] presented a new algorithm for frequent itemset mining that uses a different data structure called Inverted Index Structure (IIS). The algorithm requires to scan the database only once and rescanning is not required even if minimum support threshold changes. It employs a more efficient use of the extendable-
itemset property to reduce the number of steps of mining. The IIS-mine performs much better
for dense datasets than FP-tree in terms of run-time, memory consumption and scalability
when the minimum support threshold is varied.

M. El-Hajj and O. R. Zaiane [HAJ2003] coined the COFI-tree approach, a divide and
conquers approach, which do not seek to find all frequent patterns at once, but independently
find all frequent patterns related to each frequent item in the frequent 1-itemset. The main
differences between COFI-approach and the FP-growth approach are the followings: (1) It
only build one COFI-tree for each frequent item $A$. This COFI-tree is non-recursively
traversed to generate all frequent patterns related to item $A$. (2) Only one COFI-tree resides in
memory at one time and it is discarded as soon as it is mined to make room for the next
COFI-tree.

2.3.4 Graph Based Approaches of Rule Mining

Graphs has become increasingly popular in modeling complex structures like biological
structures, circuits, images, protein structures and chemical compounds. Graph theory has
also been successfully applied in data mining. Several approaches based on graphs have been
introduced that mine data efficiently.

Yen and Chen [YEN2001] proposed graph based algorithm, DLG to efficiently solve the
problem of mining association rules. It can construct directed graphs demonstrating the
relationship between itemsets, traverse through the directed graphs to generate frequent
itemsets, and adopt the matrix storage technology. It does not produce candidate itemsets and
only requires scanning the database one time. The experiment has already proved that DLG
algorithm has better performance than many frequent itemsets discovery algorithm. But there
will be a large number of redundant computing in DLG when there exist many short pattern itemsets. The authors discussed a graph-based association rule mining algorithm which directly generates frequent candidate itemsets through constructing directed graph to form association rules. But this algorithm occupies a great deal of time for checking the candidate itemsets and thus an improved algorithm was proposed. The improved algorithm utilizes the method of judging the nodes to cut the redundant nodes of candidate itemsets. With smaller minimum support, the improved algorithm discovers the frequent itemsets efficiently.

Inokuchi A. et al. [INO1998] presented a novel approach namely AGM to efficiently mine the association rules among the frequently appearing sub-structures in a given graph dataset. A graph transaction is represented by an adjacency matrix and the frequent patterns appearing in the matrices are mined through the extended algorithm for basket analysis. The algorithm has been proved to be efficient on several real and artificial datasets.

Anurag Choubey et al. [CHO2012] proposed a graph based approach for frequent pattern mining. They introduced a graph structure that captures only those itemsets that needs to define a sufficiently immense dataset into a submatrix representing important weights and does not give any chance to outliers. They devised a strategy that covers significant facts of data by drilling down the large data into a succinct form of an adjacency matrix at different stages of mining process. The graph structure is so designed that it can be easily maintained and the trade off in compressing the large data values is reduced. Experimental results proved the effectiveness of the proposed approach over traditional approaches.

Chung-Wen Cho et al. presented a graph-based approach for mining of inter-transaction association rules. The author classified the patterns in three types and respectively discussed
intra-type and inter-type relationship. Moreover, they show the ways to transform problems in either same or different types. The suggested approach needs to scan the database twice. Two algorithms are proposed namely, S-K algorithm and M-L algorithm. The graph-index and the E-matrix are employed to enhance their efficiency.

2.3.5 Advanced Approaches of Rule Mining

Ashrafi et al. [ASH2004] discussed the issue of redundant association rules. In their work, several methods to eliminate redundant association rules have been presented. Also, methods have been provided to produce small number of rules from any frequent or frequent closed itemset generated. The author presented additional redundant rule elimination methods that first identify the rules that have similar meaning and then eliminate these rules. However, the method never drops any high confidence or interesting rule from the rule set.

Jaroszewicz and Simovici [JAR2002] addressed the problem of redundant rules using maximum entropy approach. The problem of efficiency of maximum entropy computation is handled by using closed form solutions for the most frequent cases. Analytical and experimental evaluation of their proposed technique indicates that it performs much better than other techniques and produces a small set of interesting association rules.

Techapichetvanich and Datta [TEC2004] pointed out the fact that association rule mining task is done more efficiently if some proper visualization tool is used during mining of association rules. They presented a three-step visualization approach for mining market-basket association rules. These three steps include discovering frequent itemsets, mining association rules and finally visualizing the mined association rules.
Omiecinski [OMI2003] suggested the use of other measures of interestingness for association rules. The author takes a different vision of significance and recommended the use of all-confidence and bond measure of interestingness. These measures are indicators of the degree to which items in an association are related to each other. With all-confidence, an association is considered interesting if the all rules that can be produced from that association have a confidence greater than or equal to a minimum all-confidence. Bond is like support value but it is applicable to a subset of the data rather than entire data set.

Liu et al. [LIU1999] discussed the problem of association among rare items. They talked about the scenario that occurs while mining rules that involve rare or infrequent items. The support value is kept low. But a low support value will generate many rules that are of no interest. Use of high support will produce less rules but will eliminate rules with rare items. The authors tackle this problem by allowing users to specify different minimum support for various items in their algorithm.

Brin et al. [BRI1997a] introduced the notion of negative association rules. The model used by them is chi-square based. To determine the nature of relationship (negative or positive) they make use of correlation metric. The researchers attack two key problems in negative association rule mining viz. how to effectively search for interesting itemsets and how to effectively identify negative association rule of interest.

Savasere et al. [SAV1998] presented a novel idea for mining strong negative rules. The authors combine positive frequent itemsets with domain knowledge in the form of taxonomy to mine negative associations. But the approach is difficult to generalize because it is domain dependent. A similar approach was proposed by Zhang C. et al. [ZHA2004]. They derived a
new algorithm for generating both positive and negative association rules. For better pruning of the frequent itemset generated, another measure called mininterest has been added on the top of support-confidence framework.

2.3.6 Techniques that Enhance Efficiency of Rule Mining

Yuan Y. [YUA2005] suggested the use of matrix for mining association rules. The Matrix algorithm generates a matrix with entries 0 and 1 by scanning the 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. Experimental results confirms that the proposed algorithm compete with the Apriori algorithm and is more efficient in terms of run time and complexity.

Wang and Tjortjis [WAN2004] presented another efficient algorithm called PRICES. The approach reduces large itemset generation time, which is the most time consuming step, by scanning the database only once and using logical operation in the process.

Toivonen [TOI1996] proposed a sampling approach for mining association rule mining. The approach works in two phases. During first phase a sample of database is obtained and all associations in the sample are generated. Then in the second phase the results are validated against the entire database. The author used a lowered minimum support on the sample. As the approach is probabilistic all possible rules cannot be found in the first phase. Some associations may be deemed infrequent in the sample but may actually be frequent in the entire database.
Parthasarathy [PAR2002b] presented an efficient method to progressively sample for association rules. His approach relies on a novel measure of model accuracy. The approach is based on the identification of a representative class of frequent itemsets that simulate accurately the self-similarity values across the entire set of associations and an efficient sampling methodology that hides the overhead of obtaining the progressive sample by overlapping it with useful computation.

Chuang et al. [CHU2005] discussed another progressive sampling algorithm which focus on the identification of accurate sample size for mining association rules. The algorithm was named as SEE (Sampling Error Estimation). The method is good because an appropriate sample size can be determined without the need of executing association rules. Also the identified sample size is highly accurate thus the association rules can be much efficiently executed on a sample of this size to obtain sufficiently accurate results.

Li and Gopalan [GOP2004] reported in their work the process of deriving the sufficient sample size based on central limit theorem for sampling large datasets with replacement. They discussed the key issues while using sampling in data mining viz. how to sample the data and how big the sample should be for a given error bound and confidence level. They also stated that if data is coming at a faster rate than can be processed, sampling seems to be the ultimate choice.

Cheung et al. [CHE1996] presented an algorithm called FDM. This is a parallelization of Apriori to shared nothing machines, each with its own partition of database. The author suggested performing database scans on local partition independently at every level and on each machine. Another Apriori based distributed ARM algorithm named DDM is proposed
by Schuster and Wolff [SCH2001]. Like FDM candidates in DDM are generated levelwise and 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.

Cheung et al. [CHE1998] proposed another efficient parallel mining algorithm FPM (Fast Parallel Mining) for shared-nothing parallel systems. It follows the count distribution approach and has coined two powerful pruning techniques viz. distributed pruning and global pruning. It adheres to a simple communication scheme that performs only one round of message exchange in each iteration.

Wojciechowski and Zakrzewicz [WOJ2002] focus on improving the efficiency of constraint-based frequent pattern mining by using a dataset filtering technique. The dataset filtering conceptually transforms a given data mining task into an equivalent one operating on a smaller dataset. The authors presented transformation rules for various classes of patterns: itemsets, association rules and sequential patterns and discussed implementation issues regarding integration of dataset filtering with well known pattern discovery algorithms.

Das A. et al. [DAS2001] introduced Rapid Association Rule Mining (RARM) that uses tree structure to represent the original database and avoid candidate generation process. For improving the efficiency of existing mining algorithms, the author suggested to apply constraints during the mining process to generate only those association rules that are interesting to the users instead of all rules.

The huge literature related to data mining techniques has been surveyed. Since the inception of the technology, researchers are involved either in devising new approaches for mining
useful information or inventing new ways for enhancing the efficiency of the existing approaches.

Another theme of the current research work is swarm intelligence. This is also a well researched area that is a subfield of artificial intelligence related with the research on the behavior of biological swarms and the natural intelligence found in different species of birds, ants, fishes, herds and bees. The literature related to swarm intelligence is discussed in the next section.

2.4 Swarm Intelligence

The two mainstreams of swarm intelligence area are: Ant Colony Optimization [DOR2004] and Particle Swarm Optimization [KEN1995]. Since last two decades these techniques have spread their influence in almost all areas of optimization. Several variants and methods of these techniques have been devised over time. The popular applications of these techniques are discussed in next sub-sections.

2.4.1 Ant Colony Optimization

The Ant Colony Optimization (ACO) meta-heuristic is inspired by the foraging behavior of ants. The ants’ goal is to find the shortest path between a food source and the nest. Each path constructed by the ants represents a potential solution to the problem being solved. ACO has also been used in applications such as rule extraction, Bayesian network structure learning, and weight optimization in neural network training.

Dorigo et al. [DOR1991] introduced the term ACO and presented a framework for the use of the technique in different areas. The specific requirements and general steps that must be
adhered for applying the ACO based approached were coined. The idea of utilizing the forging behavior of ants for solving complex optimization problems was presented. Ants communicate indirectly through modifying the environment and use a feedback mechanism to attract other ants. As more and more ants follow a trail, the chances of adopting the trail by more upcoming ants increases and this ultimately leads to convergence of the solution. The same idea is applied in artificial ant system for obtaining optimal solution.

Dorigo et al. [DOR1996] proposed an algorithm Ant System for solving the travelling salesman problem. This NP-complete problem focuses on finding the shortest path that visit each city exactly once. ACO has been successfully applied for solving TSP. Although the performance of Ant System did not compete with the traditional algorithm, it is considered to be the basis of many other ACO algorithms that achieve better performance for many applications.

Stützle and Hoos [STÜ1997] proposed an improvement over the Ant System namely Max-Min ant system. This approach makes a balance between a better exploitation of the best solution and an exploration of the solution space effectively avoiding early stagnation. The Max-Min system differs from traditionally proposed ant system in three ways. First after each iteration only the best ant is allowed to add pheromone. Secondly, to avoid early stagnation, the range of possible pheromone trails is limited to an interval \([\tau_{min}, \tau_{max}]\). Finally the initial pheromone value of each trail is set at \(\tau_{max}\).

Gambardella and Dorigo [DOR1997a] presented Ant Colony System (ACS) that exploits the search experience accumulated by the ants more strongly than Ant System. Also pheromone evaporation and pheromone deposit take place only on the arcs belonging to the best-so-far
tour. The ants remove some pheromone from the arc to increase the exploration of the alternative paths.

The Rank-based method is another improvement over AS and was proposed by Bullnheimer et al. [BUL1999]. In this method, the amount of pheromones decreases with rank of ants. Before updating the pheromone trail, the ants are sorted by increasing tour length and the quantity of pheromone is weighted according the rank of the ant. In each iteration only \((w-1)\) best ranked ants and the ants that produced the best-so-far tour are allowed to deposit pheromone.

2.4.2 Particle Swarm Optimization

The PSO meta-heuristics is motivated by the coordinate movement of fish schools and bird flocks. The PSO is compounded by a swarm of particles. Each particle represents a potential solution to the problem being solved and the position of a particle is determined by the solution it currently represents.

A first application of PSO to classification is reported by Sousa et al. [SOU2004]. A sequential covering algorithm is employed, with each rule being a conjunction of terms. Rule quality is measured as the product of sensitivity and specificity, just as with AntMiner.

In the work reported by Alatas and Akin [ALA2009], rule mining is considered as a multi-objective optimization problem with predictive accuracy and comprehensibility objectives. Each rule is evaluated according to multiple rule evaluation criteria: accuracy and comprehensibility. The chaotic PSO algorithm searches for Pareto optimal classification rules.
2.4.3 Other Swarm Intelligence Techniques

Karaboga, D. [KAR2005] discussed Bee foraging. Honey bee colonies have a decentralized system to collect the food and can adjust the searching pattern precisely in order to enhance the collection of nectar. Bees can estimate the distance from the hive to food sources by measuring the amount of energy consumed when they fly besides the direction and the quality of the food source. This information is shared with their nest mates by performing a waggle dance and direct contact.

Passino, K. [PAS2002] introduced Bacterial foraging (BFO) and applied the algorithm to the optimization of a benchmark function. In reproduction, the health of each bacterium represents its fitness value. All bacteria are sorted according to their health status and only the first half of population survives. The surviving bacteria are split into two identical ones in order to form a new population. Thus the population of bacteria is kept constant.

Yang, X.S. [YAN 2009] discussed the behavior of Lampyridae bioluminescence. Lampyridae is a family of insects that are capable to produce natural light (bioluminescence) to attract a mate or a prey. They are commonly called fireflies or lightning bugs. The firefly algorithm (FA) was applied to the optimization of benchmark functions.

Havens, T. et al. [HAV2008] presented RIO algorithm. The cockroaches’ agents are defined using three simple behaviors: cockroaches search for the darkest location in the search space and the fitness value is directly proportional to the level of darkness (find darkness phase); cockroaches socialize with nearby cockroaches (find friend phase) and third behavior is cockroaches periodically become hungry and leave the friendship to search for food (find food phase).
Yang, X. [YAN2010] proposed a Bat Algorithm (BA), based on the echolocation behaviour of bats. It is potentially more powerful than particle swarm optimization and genetic algorithms. The primary reason is that BA uses a good combination of major advantages of these algorithms in some way. In these studies, it is needed to carry out the sensitivity analysis, to analyze the rate of algorithm convergence, and to improve the convergence rate even further.

Eusuff, M.M. et al. [EUS2003] proposed the shuffled frog-leaping algorithm which is a memetic metaheuristic that is designed to seek a global optimal solution by performing a heuristic search. It is based on the evolution of memes carried by individuals and a global exchange of information among the population. it combines the benefits of the local search tool of the particle swarm optimization and the idea of mixing information from parallel local searches to move toward a global solution.

The wide research in the area of swarm intelligence has fascinated the researchers for making use of these techniques in data mining too. The next section throws light on the applications of these techniques of ACO in data mining.

### 2.5 Data Mining with Ant Colony Optimization

Ant colony optimization technique has been recently applied to data mining tasks. The most prominent use of ACO in data mining is in classification rule discovery. Clustering has also scope for use of ACO, especially brood sorting technique and many researchers applied these methods for clustering tasks. However use of ACO for association rule mining is limited. The most notable researches have been discussed below:
2.5.1 Classification Rule Discovery with AntMiner

Data mining consists of several tasks and ACO has been applied for many of these. Classification is an important data mining task, where the value of a discrete (dependent) variable is predicted based on the values of several independent variables. ACO within the data mining community has been used primarily for supervised classification. Although ACO has been primarily used for clustering, the bulk of the research addresses the area of classification rules discovery. The most notable research in this direction is the AntMiner rule induction technique. AntMiner was the first application of ACO to classification reported by Parpinelli et al. [PAR2001] [PAR2002].

The AntMiner+, a successor of AntMiner proposed by Martens et al. [MAR2007] differs from the AntMiner version in several ways. The environment is defined as a directed acyclic graph (DAG), so that the ants can choose their paths more effectively when the AntMiner environment is fully connected.

AntMiner2 [LIU2002] further extended AntMiner but used a simpler, though less accurate density estimation equation as compared to the heuristic value. This made AntMiner2 computationally less expensive without a significant degradation of the stated performance.

AntMiner3 [LIU2003] also extended AntMiner by introducing two major changes, resulting in increased accuracy. Firstly, a different update rule is used which is defined with the quality of a rule set to the sum of its sensitivity and specificity. Secondly, more exploration is encouraged by means of a different transition rule that increases the probability of choosing terms not used in previously constructed rules, as implemented by the Ant Colony System.
Finally, it is worth noting that an extension of AntMiner that generates fuzzy rules has also been proposed.

Recently, a continuous version of the ACO metaheuristic has been proposed, aimed at solving continuous optimization problems by Socha et al. [SOC2004]. Instead of using a probability function that has one value per node (denoting a single value for the discrete variable), the ACO$_R$ version employs a probability distribution consisting of several Gaussian probability density functions that can easily be sampled.

Instead of a typical rule induction approach, a rule extraction approach is proposed by Özbakır et al. in their TACO-miner algorithm. Rule extraction techniques from neural networks or support vector machines extract rules that mimic these black box models as closely as possible [OZB2009].

A first application of PSO to classification is reported by Sousa et al. A sequential covering algorithm is employed, with each rule being a conjunction of terms. Rule quality is measured as the product of sensitivity and specificity, just as with AntMiner [SOU2006].

In Holden and Freitas [HOL2008], a combination of both ACO and PSO is used to build rule-based classification models that can handle both nominal and continuous variables. While for PSO implementations nominal variables need to be encoded in binary variables and ACO implementations require discretization, the PSO/ACO$_2$ technique directly deals with both types of variables. This technique extends and refines the first version by the same authors [HOL2008] [HOL2005].
In the work reported by Alatas and Akin rule mining is considered as a multi-objective optimization problem with predictive accuracy and comprehensibility objectives. Each rule is evaluated according to multiple rule evaluation criteria: accuracy and comprehensibility. The chaotic PSO algorithm searches for Pareto optimal classification rules [ALA2009].

Souvik Sengupta et al. have used the technique of ACO for construction of learning path for e-learning. They have taken the data mining based frequent pattern graph model to define the association and sequencing between the words and then adopted the Ant Colony Optimization, to derive a searching technique to obtain an efficient and optimized learning path to reach to an unknown term [SOU2011].

Veenu Mangat has worked in the field of rule mining for medical domain. She has reported that medical domain generates a huge quantity of information daily. So extracting useful information and providing help for scientific decision making, for the purpose of diagnosis and treatment of disease, has become necessity. The author has presented various techniques for rule mining in medical domain, identified gaps and proposed a novel hybrid framework for efficient rule mining [MAN2012]. In another paper, the author has reported the use of swarm intelligence for classification rules in medical domain and proposed a combined ACO/PSO approach for extracting classification rules [MAN2010].

2.5.2 Clustering with ACO

All previously discussed data mining techniques use ACO for classification. The ACO metaheuristic can also be applied to the clustering task. ACO metaheuristic is based on the foraging principles of ants, but clustering algorithms have been introduced that mimic the
sorting behavior of ants. It has been shown that several ant species cluster dead ants in so-called cemeteries to clean up the nest [BON1999].

The data clustering algorithm proposed by Lumer and Faieta called LF algorithm is based on the same ant behavior as modeled by Deneubourg et al. Additions to this basic LF algorithm that have been proposed are an adaptive setting of the parameters, allowing multiple data items to be transported at once [LUM2004].

An improved version of the LF algorithm has been proposed, named ATTA (Adaptive Time-dependent Transporter Ants), which has given its reported results and rather extensive changes. ATTA has two variants: one which is limited to a topographic mapping, named ATTA-M, and one which actually results in clusters of data (ATTA-C). The main benefits of the ATTA-C technique is an explicit partitioning without need of human intervention and a prior setting of parameters depending on dataset characteristics [HAN2006].

Van der Merwe and Engelbrecht applied PSO for data clustering. Each solution is represented by the coordinates of a user pre-defined number of cluster centroids. Each data instance is assigned to the nearest cluster centroid using Euclidean distance. The fitness function used is the quantization error, which can be seen as the average distance from a data point to its cluster centroid, averaged over the different clusters [VAN2003].

2.5.3 Association Rule mining with ACO

Vijayarani and Sathiya Prabha [VIJ2011] have reported the association rule hiding problem. The authors have proposed Artificial Bee Colony Optimization algorithm for hiding the
sensitive association rules and analyzed the efficiency of the Artificial Bee Colony Optimization technique by using various performance factors.

Sudha Sadasivam et al. have reported in their work [SAD2012] that privacy preserving data mining hides the sensitive rules and prevent the data from being disclosed to public. They have proposed two approaches to hide sensitive fuzzy association rules; first, decreasing support value of item in RHS of association rule for sensitive rule hiding by fuzzification using DSR and second, sensitive rule hiding using PSO.

T. Karthikeyan and J. Mohana Sundaram [KAR2012] have reported in their work the use of ACO with association rules. The authors have conducted experiment with the Antminer and Antminer+ algorithm for mining association rules from the soil dataset and discussed the results. These algorithms are basically used for classification rule discovery, but the authors have applied it for association rules.

R.J. Kuo and C. W. Shih [KUO2007] discussed the association rule mining problem using ant colony system and applied the technique for mining the national health insurance research database. The study has demonstrated that the novel approach deals with the discovery of hidden knowledge very well. The results of the study presented several interacting relationships among the disease items. The proposed algorithm proved to be much better than Apriori both in terms of efficiency as well as reliability according to expert questionnaire survey results. In addition, the proposed method allows the users to specify the search constraints which make the extracted rules more appropriate to user’s needs and faster computational speed.
2.6 Summary

This chapter presents the taxonomy of techniques of data mining based on the extensive survey of literature for a couple of decades. The various approaches of association rule mining and swarm intelligence are critically analyzed. The benefits and shortcoming of the surveyed approaches are summarized as under:

Majority of approaches for association rule mining [AGR1993] [BRI1997] [PAR1995] [LIN1997] that follow the horizontal database performs repeated scans of the database. This consumes a lot of time and efforts. These approaches generates a large number of candidate sets and the support counting for these candidate sets is a complex task as the whole transaction database has to be matched with the itemset whose support is to be calculated. In addition to this large storage space in memory is required for storing the intermediate itemsets. Thus if the database is large, the time and space requirements becomes unmanageable.

The approaches that follow the vertical layout of database [ZAK1997] [ZAK2003] [KAN2010] results in reduction of repeated scans of database and hence reduce the total time elapsed in generating association rules. These approaches use intersection approach for calculating the support count of the items and hence instead of scanning the database again and again, these approaches apply simple logical operations. But large number of candidates is still a problem.

The approaches that use projected layout [HAN2000a] [PRA2000] [SHA2010] [SAH2012] are a real improvement over others. These approaches works by converting the database in some compact form like tree or graph and then perform data mining on the compact database.
This results in better performance but the conversion of transaction database in some other form takes extra efforts and demands for large storage space.

The swarm intelligence approaches have been applied in data mining tasks specifically in classification and clustering. The use of these approaches in association rule mining is negligible. However some researchers [KUO2007] [VIJ2011] [SAD2012] [KAR2012] proposed the use of ACO and PSO for association rule mining but they inherently use classification rule approach. Therefore in the present research work, an attempt has been made to make use of ACO for generating association rules. Novel approaches are proposed that generate frequent itemsets and association rules efficiently. The approaches are evaluated by implementing them in Microsoft Visual Studio 2010 with C# platform and the results are compared with the results of other approaches that have been accepted as a standard.

In Chapter 3 next, the detailed description of the Association Rule Mining technique has been provided. Different approaches for mining association rules are discussed with pros and cons of each approach. A novel graph based approach for finding frequent itemsets is proposed in the chapter and the performance of the approach is evaluated and compared with existing approaches.