IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 18, Issue 4, Ver. VI (Jul.-Aug. 2016), PP 11-17
www.iosrjournals.org

An Analytical Study of Genetic Algorithm for Generating Frequent Itemset and Framing Association Rules At Various Support Levels

D. Ashok Kumar[1], T. A. usha[2]
Department of Computer Science and Applications Government Arts College, Tiruchirappalli – 620022, Tamil Nadu, India.

Abstract: *In customary, frequent itemsets are propogated from large data sets by employing association rule mining algorithms like Apriori, Partition, Pincer-Search, Incremental and Border algorithm etc., which gains inordinately longer computer time to cast up all the frequent itemsets. On utilizing Genetic Algorithm (GA) the scheme is reformed.. The outstanding benefit of utilizing GA in determining the frequent itemsets is to discharge exhaustive survey and its time convolution subsides in collation with other algorithms, since GA is built on the greedy mode. The effective plan of this report is to detect all the frequent itemsets and to generate the association rules at various levels of minimum support and confidence defined by the user, with very less time and less memory from the furnished data sets using genetic algorithm.*

Keywords: *Genetic Algorithm (GA), Association Rule, Frequent itemset, Support, Confidence*.

I. Introduction

In recent trends, data mining plays a vital role for framing association rules among the massive collection of itemsets. Frequent itemsets are framed from large data sets by making use of association rule mining, takes ample of time to figure out all the frequent itemsets. By utilizing the Genetic Algorithm (GA) we improved the results of association rule mining. Genetic Algorithms are authoritative and broadly useful stochastic search and optimization methods based on the concepts of natural selection and natural evaluation. On the whole, the objective of this report is to observe the entire frequent itemsets and also generates the association rules in a short span of time from the large datasets by using the proposed genetic algorithm, GAM**.**

1.1 Association Rule Mining

Association rule mining technique was set forth in 1993 by Agrawal et al.[7], who developed Apriori algorithm to unfold the ARM based situations. It furnishes information of the form of "if-then" statements. Association rule mining approach discovers significance relationship and correlations among data set.

Assume I = {i1, i2, . . . , im} be a set of verbatim, called items. The execution in D bears a single transaction ID and comprises a subset of the items in I. X => Y, where X, Y ⊆ I and X∩Y [1]. For the axiom, X=>Y, X is the antecedent (left-hand-side or LHS) and Y is the consequent (right-hand-side or RHS) respectively.

Association rules basically comprises of two salient features namely, support(s) and confidence(c) [8]. Since the database is bulky and users interested about only those repeatedly acquired items. Minimal support and minimal confidence are the two different thresholds predefined by the users to frame the required association rules.

1.2 Frequent Itemset Mining

An k-itemset that is composed of k items from I, is frequent if it is present in the Transaction (T) not smaller than s times, where s is the user-designated minimum support threshold and s <= n.

1.3 Genetic Algorithm

In 1975, John Holland commenced the Genetic Algorithm at University of Michigan. Genetic Algorithm is a versatile heuristic penetrating algorithm constructed from the evolution of natural selection and genetics. This probing algorithm is constructed on the techniques of biological evolution. In the year 1992, John Koza employed Genetic Algorithm to yield the programs to execute reliable tasks and entitled as Genetic Programming. It is also a module of evolutionary computing. Genetic algorithms are simulated by Darwin's theory about the evolution, expressed as "Survival of the Fittest". It generates a result by fusing selection, recombination and mutation. It randomly searches the dataset to resolve the appropriate problems. It signifies that the more desirable results emit from the earlier origination until a close by ideal result is achieved. It provides streamlined, valid procedure for the most appropriate and machine learning applications [2]. Genetic

algorithm is a stated procedure that symbolizes its possible result as stream of genes termed as Chromosomes. A set of creatures (Chromosomes) called inhabitants.

In each repetition of the algorithm, the population is altered. Genetic Algorithm's iterations are called as generations. Standard Genetic algorithm applies genetic operators such as selection, crossover and mutation. It generates solutions for successive generations. The algorithm is stopped by obtaining an optimum solution [13].
The operators of genetic algorithm are as follows: **Selection**: According to Darwin's evolution theory, the chromosomes with higher fitness ratings are selected from the population to be the parents to crossover that should survive and create new offspring.

Crossover: It leads to effective combination of schemata (sub solutions on different chromosomes). It implies by selecting an arbitrary location in the string and swapping the components either to the right or to the left of this position with some other string subdivided correspondingly to construct two neonatal offspring.

Mutation: After a crossover is performed, mutation takes place. It is an arbitrary change in a position. It prevents the algorithm from immovable position. The procedure changes a 1 to a 0, or a 0 to a 1. This alteration is formed with a very low probability.
First, genetic algorithm produces an initial population of individuals, and then evaluates the fitness of all individuals. The following process continues until the optimal solution met. First, it selects fittest individuals for reproduction. Secondly, it recombines between individuals. Then mutate the individuals and then evaluate the fitness of the modified individuals to generate a new population [6].

Step 1: Choose the initial population of individuals
Step 2: Establish the fitness of every individual in that population.
Step 3: Redo on this generation until the end-point: (time limit, sufficient fitness achieved, etc.)
a) Select the best-fit individuals for reproduction.
b) Reproduce new individuals through crossover and mutation operations.
c) Evaluate the individual fitness of new individuals.
d) Replace least-fit population with new individuals.

II. Related Works

2.1 Genetic Algorithm used in the Association Rules

Wakabi-Waiswa, P.P., et al., proposed [12] "*Generalized Association Rule Mining Using Genetic Algorithms*". In this paper, Association rule mining is designed for the Genetic Algorithms(GAM) at various levels of minsupport. It yields very fast results. It generalized a very large database of transactions, where each transaction contains a set of items, and a classification on the items, and then the associations between items at any level of the classification have been found. It improves the performance based on the large number of itemset and high level of minimum support.

Ghosh S, Biswas S, Sarkar D and Sarkar P.P, "*Mining Frequent Itemsets Using Genetic Algorithm*", proposed [6] the algorithm to find frequent itemsets using genetic algorithm. The association rule mining algorithm like apriori, partition, fp-tree, etc., generate the frequent itemsets. However, it takes too much time to compute the frequent itemsets. The main aim to introduce genetic algorithm is to reduce the computing time. Genetic algorithm performs global search to generate the frequent itemsets. The time complexity and memory usage is less when compared to the association rule mining algorithm because the genetic algorithm is constructed by the greedy approximation.

This paper compares and analyzes the genetic algorithm for finding the frequent itemsets at different measures of support level. The proposed Genetic algorithm GAM uses fitness function:
$f(i) = {}_{i=1,j=1}\Sigma^{ni,nt}\ P(i,j) * Sup(j)$ for selecting the samples in each stratum. This algorithm identifies the frequent itemsets repeatedly using the following steps.
First, the sample data is selected from the datasets like contextPasquier99, Mushroom.dat and pumsb_star.dat.
Second, Fitness is calculated for each individual by using the formula: $Fitness(i) = {}_{i=1,\ j=1}\Sigma^{i=nt,\ j=ni}.P(i,j)*Support(j)$
Thirdly, Roulette Wheel selection method is used to select the individuals from the parents to be involved in recombination.
Fourthly, new individuals can be created by using the genetic operators such as crossover and mutation.
Finally, some of the new individuals are replaced with their parents.
Dou W, Hu J, Hirasawa K and Wu G, "Quick Response Data Mining Model Using Genetic Algorithm",[3] provided a base for this paper to find the maximal frequent itemsets using Genetic algorithm.
In this paper, GAM algorithm is expanded to deal with generalized association rules.

III. Methodology

In this paper, the GAM is employed on huge data sets like contextPasquier99, Mushroom.dat, and Pumsb-Star.dat, to explore the recurrent itemsets. We first load the sample of records from the transaction database that fits into memory. The genetic learning starts as follows. An initial population is created consisting of randomly generated transactions. Each transaction can be represented by a string of bits. Our proposed genetic algorithm based method for finding frequent itemsets, repeatedly transforms the population by executing the following steps:

(1) Fitness Evaluation: The fitness (i.e., an objective function) is calculated for each individual.

(2) Selection: Individuals are chosen from the current population as parents to be involved in recombination.

(3)Recombination: New individuals (called offspring) are produced from the parents by applying genetic operators such as crossover and mutation.

(4) Replacement: Some of the offspring are replaced with some individuals (usually with their parents). One cycle of transforming a population is called a generation. In each generation, a fraction of the population is replaced with offspring and its proportion to the entire population is called the generation gap (between 0 and 1).

Let P= {P1, P2, …, Pn} be the set of products and T={T1, T2, …, Tn} be the set of transactions in the database for framing association rules. Every transaction Ti has its own ID and holds a subset of the items in I, called itemset. An association rule is defined as X->Y, where X U Y is subset of I and X∩Y = Ø [10][9].

3.1 Identifying Best Association Rules and their Optimization Using Genetic Algorithm

An itemset can be a single item (e.g. mineral water) or a set of items (e.g. sugar, milk, red tea). The two significant standards of association rules are support(S) and confidence(C) [6]. Support(S) of an association rule is the percentage of transactions in the database that contain the itemset X U Y. Confidence (C) of an association rule is the percentage/fraction of the count of executions that comprises X U Y to the total number of records that contain X. Confidence factor of X->Y can be defined as:

$C(X \rightarrow Y) = S(X \cup Y) / S(X)$.

In this paper, the genetic algorithms focus on multi-objective problem [4], [5] with emphasis on association rule mining algorithm. Genetic algorithm is restated methodology, suitable for huge and heterogeneous search space and optimal situations. The following points are considered for utilizing genetic algorithm: Encoding/decoding schemes of chromosomes, Population size, Fitness value, Selection, Crossover and Mutation.

IV. Algorithmic View

4.1 Existing Algorithm GAM:

The function of extracting association rules over the selected data is identified as essence of proficiency of locating the trends. Association rule mining furnishes a valuable procedure for detecting relationship in the group of items associated with the consumer dealings. In general, the association rule is indicated as X=>Y, where X is the antecedent and Y is the consequent. Association rule counts the appearance of Y with respect to the existence of X, determined by the support and confidence value.

Frequent itemset: Assume A to be the set of items, T as the transaction database and σ as the user specified minimum support. An itemset X in A (i.e., X is a subset of A) is denoted as a frequent itemset in T with reference to σ, if support(X) T >= σ.

Extracting association rules can be fragmented into two sub-problems as mentioned below:

1. Generating all itemsets that measures support higher than, or as same as the user specified minimal support. That is, generating all large itemsets.
2. Generating all the rules that have minimum confidence.

We can generate the association rule with more than one number of consequent items is generated by the following method:

1. Find the rule in which number of consequents =1.

2. For the given rules $p(x \rightarrow y)$ and $p(x \rightarrow z)$, the rule $p(x \rightarrow yz)$ is generated by the intersection of both the association rules and get a new rule $p(x \rightarrow yz) = p(xyz)/p(x)$.

4.2 Genetic Algorithm for feature subset selection:

The GA tool from MATLAB R2006b had been used for GA implementation. The roulette wheel is used for selection process. The crossover and mutation points are randomly generated. The applying of GA factors is highly significant. For GA parameters, if the magnitude of the population is very less, it is hard to get the best solution and for a high magnitude, the convergence time will be prolonged. Thus, the size is normally 40–60. If the crossover Pc is highly diminished, it is hard to hunt through and a Pc value is more, will abuse the individuals

with modified values. Therefore, the Pc is regularly 0.3–0.9. If the mutation value, Pm is highly diminished, it is difficult to create new individuals and a Pm value is more, it is appreciable to hunt through GA at random. Thus, the Pm is generally 0.01– 0.2. [11].

1. Select item with minimum support.
2. Find fitness using the formula Fitness(i) = $_{i=1, j=1}\Sigma^{i=nt, j=ni}$.P(i,j) * Support(j).
5. Select using Roulette Wheel selection.
6. Find frequent itemsets that satisfies the min support and min confidence.
7. Find Rules for the new population
8. Calculate the confidence of the rule by using the formula confidence = support / mean.
9. Store the rules that have min Support and minimum Confidence.
10. The fitness function for every rule x->y is acquired and the successive circumstances are probed.
11. If (fitness function > min confidence), Set B = B U {x ->y}
12. If the required propagation count is unachieved, then proceed to Step 3.
13. Stop.

The itemset that satisfies the minimum support is selected for initial transaction. Fitness is identified for each item set. Roulette Wheel selection method is adopted for selecting the sample data set. Frequent Itemset is generated, based on the minimum support. Association rules are framed for Itemsets with min confidence. Rules are stored that satisfies the minimum support and confidence.

V. Result Analysis

In this paper, it has been proposed that Genetic algorithm based solution provides the significant improvement in computational complexity. The results are analyzed and tabulated by providing several data sets like mushroom.dat, pumsb-star.dat, and contextPasquier.dat as input. The time taken and memory used at different support level is tabulated for the proposed GAM algorithm, when confidence level =50%.. It is understood from Table 1.1 a), b) and c) that GAM occupies very less memory space and consumes very less time as the size of the support level increases.

Table1.1 No. of. Rules generated, Time consumed and Memory space used

a) contextPasquier99.txt of Size 9KB with Confidence =10%

Support in %	10	20	30
No. of Rules Generated	150	108	26
Memory Space used in KB	67,604	53,306	25,644
Time Consumed	0.422632	0.228919	0.173045

b) Mushroom.Dat of Size 558KB with Confidence =50%

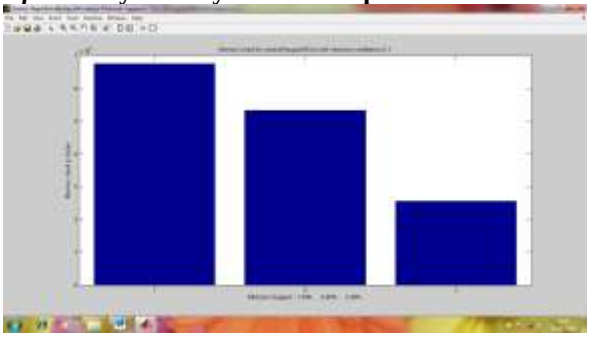
Support in %	50	60	70
No. of Rules Generated	1000	266	150
Memory Space used in KB	5,18,456	2,21,838	1,57,398
Time Consumed	2.058532	1.003202	0.779489

b) **pumsb_star.Dat of Size 11,028KB with Confidence =50%**

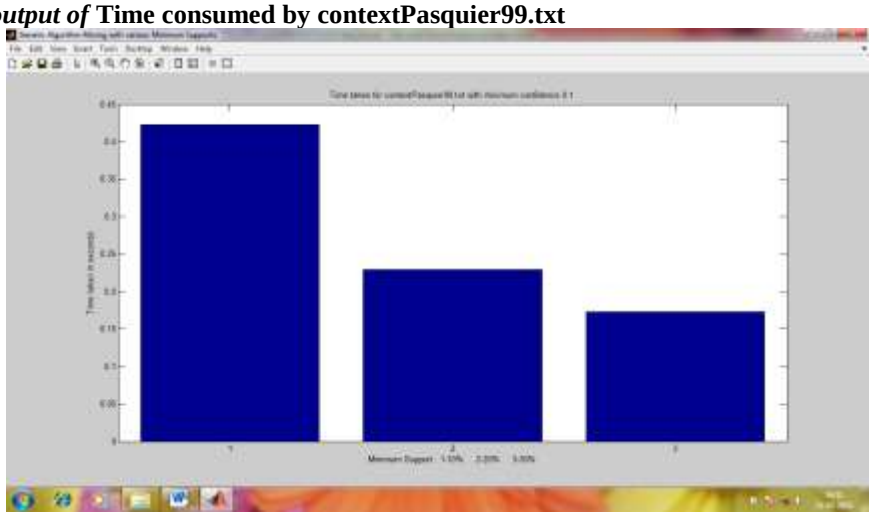
Support in %	50	60	70
No. of Rules Generated	1000	1000	92
Memory Space used in KB	21,24,362	14,24,280	8,66,238
Time Consumed	26.298496	9.292921	4.762185

The Graphical output furnishes that GAM algorithm analyses the input data and generates Frequent Itemsets and Association rules by occupying less memory space and consumption of time is minimum when the support level increases.

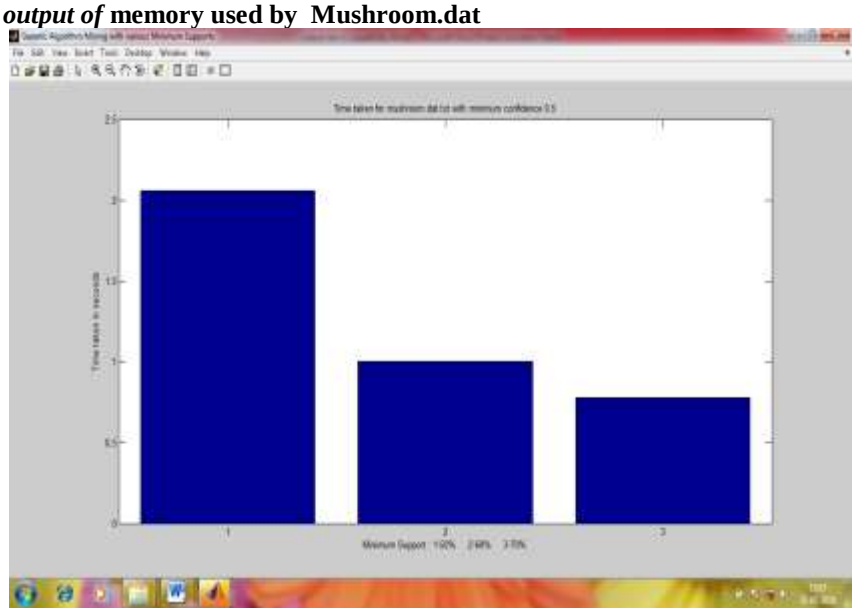
Fig 1.2 a1) Graphical output of **memory used by contextPasquier99.txt**



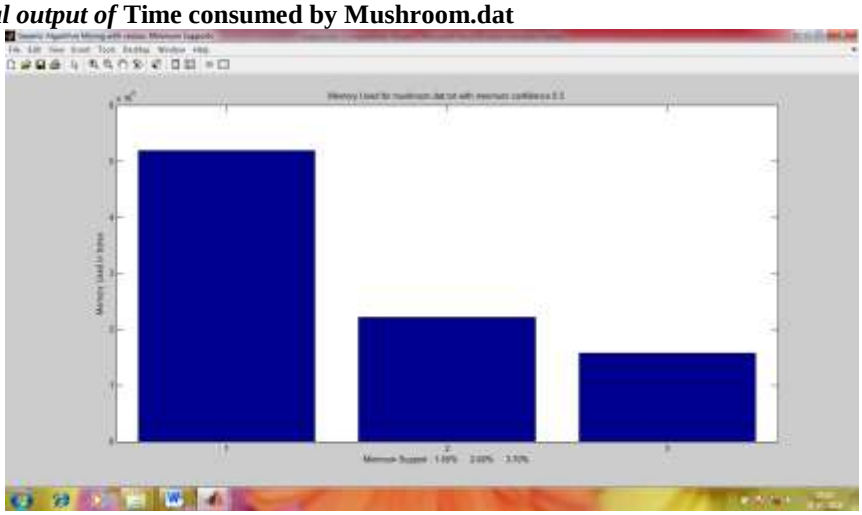
***a2) Graphical output of* Time consumed by contextPasquier99.txt**



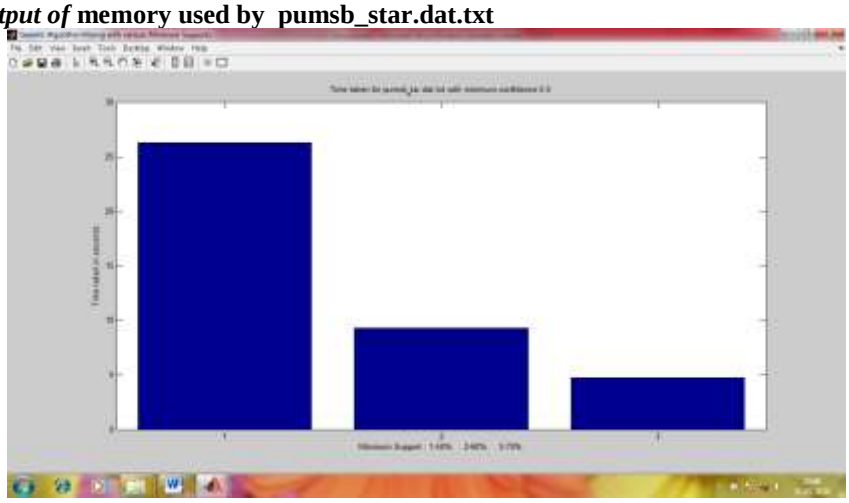
***b1)Graphical output of* memory used by Mushroom.dat**



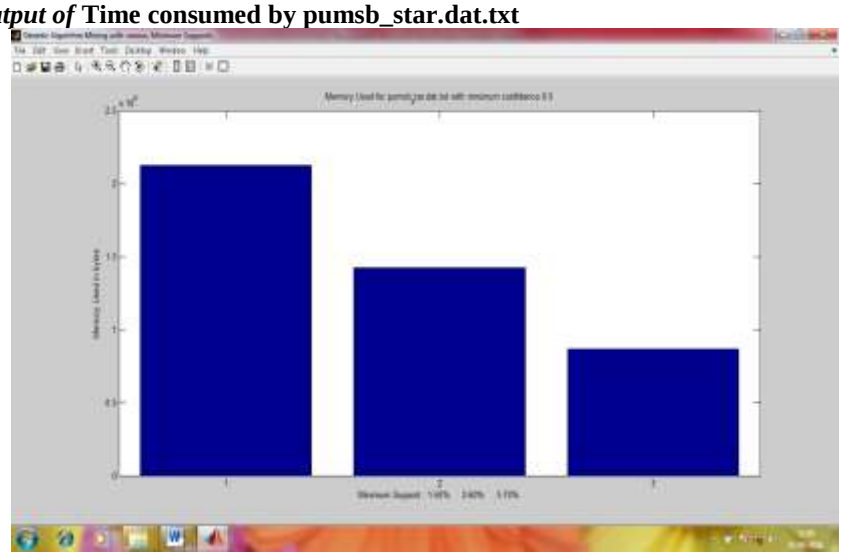
***b2)Graphical output of* Time consumed by Mushroom.dat**



***c1)Graphical output of* memory used by pumsb_star.dat.txt**



***C2)Graphical output of* Time consumed by pumsb_star.dat.txt**



VI. Conclusion

In this paper, the GAM algorithm is analyzed at various support levels for different types of data sets. The output reveals that time complexity is less and memory space occupied is also less in Genetic Algorithm.

Further, in future, it has been proposed to find the frequent itemsets using the Improved FP Tree algorithm [1] from those high quality chromosomes. This algorithm will mine the frequent itemsets with the compressed tree structure. Moreover, it can be experimented to reduce the time by using floating point data in the GAM algorithm.

References

[1]. Islam A.M.B.R. and Tae-Sun Chung, "An Improved Frequent Pattern Tree Based Association Rule Mining Technique", International Conference on Information Science and Applications, pp. 1-8, 2011.

[2]. Das S and Saha B, "Data Quality Mining using Genetic Algorithm", International Journal of Computer Science and Security, Vol. 3, No. 2, pp. 105-112, 2009.

[3]. Dou W, Hu J, Hirasawa K and Wu G, "Quick Response Data Mining Model Using Genetic Algorithm", Institute for Credentialing Excellence Annual Conference, pp. 1214-1219, 2008.

[4]. Fonesca M and Fleming J, "Multi-objective Optimization and Multiple Constraint Handling with Evolutionary Algorithms," Part I: A Unified Formulation. IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans, 28(1), pp. 26-37, 1998.

[5]. Freitas A, "Survey of Evolutionary Algorithms for Data Mining and Knowledge Discovery", Advances in evolutionary computing: theory nd applications, pp 819 – 845, 2003.

[6]. Ghosh S., Biswas S., Sarkar Dand Sarkar P.P., "Mining Frequent Itemsets Using Genetic Algorithm", International journal of Artificial Intelligence & Applications, Vol. 1, No. 4, pp. 133-143,2010

[7]. Agrawal R and Srikant R, "Fast Algorithm for Mining Association Rules," Proceedings of the 20th International Conference on Very Large Data Bases, pp. 487-499, 1994.

[8]. Kotsiantis S and Kanellopoulos D, "Association Rules Mining: A RecenOverview", GESTS International Transactions on Computer Science and Engineering, Vol. 32, No. 1, pp.71-82, 2006.

[9]. Manish Saggar, Ashish Kumar Agarwal and Abhimunya Lad, "Optimization of Association Rule Mining using Improved Genetic Algorithms"IEEE 2004.
[10]. Rupali Haldulakar and Prof. Jitendra Agrawal, "Optimization of Association Rule Mining through Genetic Algorithm", International Journal on Computer Science and Engineering (IJCSE), Vol. 3 No. 3 Mar 2011, pp. 1252-1259.
[11]. S.N. Sivanandam, and S.N. Deepa, " Introduction to Genetic Algorithms, New York: Springer-Verlag Berlin Heidelberg.
[12]. Wakabi-Waiswa P.P., Baryamureeba V and Sarukesi K, "Generalized Association Rule Mining Using Genetic Algorithms", International Journal of Computing and ICT Research, Vol. 2 No. 1, pp. 59-69, 2008.
[13]. www.cs.bgu.ac.il/~sipper/courses/ecal051/assaf-ga.ppt.

Authors

Dr. D. Ashok Kumar did his Master degree in Mathematics and Computer Applications in 1995 and completed Ph.D., on Intelligent Partitional Clustering Algorithm's in 2008, from Gandhigram Rural Institute – Deemed University, Gandhigram, Tamilnadu, INDIA. He is currently working as Associate Professor and Head in the Department of Computer Science and Applications, Government Arts College, Trichirappalli- 620 022, Tamilnadu, INDIA. His research interest includes Pattern Recognition and Data Mining by various soft computing approaches viz., Neural Networks, Genetic Algorithms, Fuzzy Logic, Rough set, etc., Cell: +91 - 9443654052.

T. A. Usha completed her M. C. A. degree , from Bharathidasan University, Tiruchirappalli, Tamil Nadu. Currently doing research in Frequent Itemset Mining Techniques and Genetic Algorithm at Bharathidasan University, Tiruchirappalli, Tamil Nadu. She is currently working as Assistant Professor in the Department of Computer Science and Applications, Government Arts College, Trichirappalli- 620 022, Tamilnadu, INDIA. Cell: +91- 9944429036.