Extraction of High Utility Itemsets using Utility Pattern with Genetic Algorithm from OLTP System

A.Saranya [1], D.Kerana Hanirex[2]

[1]PG Scholar, Dept. of Computer Science & Engg., Bharath University, Chennai, India
[2]Asst. Professor, Dept .of Computer Science & Engg. Bharath University, Chennai, India

***Abstract*:** To analyse vast amount of data, Frequent pattern mining play an important role in data mining. In practice, Frequent pattern mining cannot meet the challenges of real world problems due to items differ in various measures. Hence an emerging technique called Utility-based data mining is used in data mining processes.The utility mining not only considers the frequency but also see the utility associated with the itemsets.The main objective of utility mining is to extract the itemsets with high utilities, by considering user preferences such as profit,quantity and cost from OLTP systems. In our proposed approach, we are using UP growth with Genetic Algorithm. The idea is that UP growth algorithm would generate Potentially High Utility Itemsets and Genetic Algorithm would optimize and provide the High Utility Item set from it. On comparing with existing algorithm, the proposed approach is performing better in terms of memory utilization.

Keywords—Utility mining, High utility itemsets, UP Growth, Genetic Algorithm, Frequent itemset mining,Memory utilization

*****

I. INTRODUCTION

A. Data Mining

Data Mining refers to extracting or mining knowledge from large databases. Data mining and knowledge discovery in the databases is a new interdisciplinary field, merging ideas from statistics, machine learning, databases and parallel computing. Hence Data mining can be defined as: a)non trivial extraction of implicit, previously unknown and potentially useful information form the large databases, b)the search for the relationships and global patterns that exists in large databases but are hidden among vast amounts of data, c)refers to using a variety of techniques to identify nuggets of information or decision –making knowledge in the database and extracting these in such a way that they can be put to use in areas such as decision support , prediction, forecasting and estimation, d)it is the system self learns from the previous history of investigated system , formulating and testing hypothesis about rules which system works properly ,e)the process of discovering meaningful, new correlation pattern and trends by shifting through large amount of data stored in repositories , using pattern recognition techniques as well as statistical and mathematical techniques.[1]

For the past two decades data mining has emerged as an important research area .This is mainly due to the inter-disciplinary nature of the subject and the diverse range of application domains in which data mining based products and techniques are being employed. This includes bioinformatics, genetics, medicine, clinical research, education, retail and marketing research.

Data mining has been considerably used in the analysis of customer transactions in retail research where it is termed as market basket analysis. Market Basket Analysis is the process of exploring customer buying habits by finding associates between the different items that customers place in their “Shopping Baskets”. The discovery of such associations can help retailers develop marketing strategies by gaining insight into which items are frequently purchased together by customer.

B. Frequent Itemset Mining

Frequent itemset mining is the mining of frequent itemsets (set of items) from transactional or relational data sets. An itemset can be defined as a non-empty set of items. An itemset with k different items is termed as a k-itemset. For e.g. {bread, butter, milk } may denote a 3-itemset in a supermarket transaction .The notion of frequent itemsets was introduced by Agrawal et al [2].Frequent itemsets are the itemsets that appear frequently in the transactions. The goal of frequent itemset mining is to identify all the itemsets in a transaction dataset [3]. Frequent itemset mining plays an essential role in the theory and practice of many important data mining tasks, such as mining association rules [2,4,5], long patterns ,emerging patterns, and dependency rules. It has been applied in the field of telecommunications, census analysis and text analysis [6].

The criterion of being frequent is expressed in terms of support value of the itemsets. The Support value of an itemset is the percentage of transactions that contain the itemset.

C. Utility Mining

The restrictions of frequent or rare itemset mining inspired researchers to conceive a utility based mining approach, which allows a user to conveniently express his or her perspectives concerning the usefulness of itemsets as utility values and then find itemsets with high utility values higher than a threshold [8] .In utility based mining the term utility refers to the quantitative representation of user preference i.e. the utility value of an itemset is the measurement of the importance of that itemset in the users perspective. For e.g. if a sales analyst involved in some retail research needs to find out which itemsets in the stores earn the maximum sales revenue for the stores he or she will define the utility of any itemset as the monetary profit that the store earns by selling each unit of that itemset.

Here note that the sales analyst is not interested in the number of transactions that contain the itemset but he or she is only concerned about the revenue generated collectively by all the transactions containing the itemset. In practice the utility value of an itemset can be profit, popularity, page-rank, measure of some aesthetic aspect such as beauty or design or some other measures of user's preference.

D. Genetic Algorithm

Here basically the PHUI which are generated from previous stage has been considered as input and it would identify actual high utility items using fitness function and threshold set by user.

Chromosome:

All living organisms consist of cells[25]. In each cell there is the same set of chromosomes. Chromosomes are strings of DNA and serves as a model for the whole organism. Usually a chromosome consist of genes which is basically blocks of DNA. Each gene encodes a particular protein. Basically it can be said, that each gene encodes a trait, for example color of eyes. Possible settings for a trait (e.g. blue, brown etc) are called alleles. Each gene has its own position in the chromosome. This position is called locus.

Complete set of genetic material (all chromosomes) is called genome. Particular set of genes in genome is called genotype. The genotype is with later development after birth base for the organism's phenotype, its physical and mental characteristics, such as eye color, intelligence etc.

Reproduction:

During reproduction, first occurs recombination (or crossover). Genes from parents form in some way the whole new chromosome. The new created offspring can then be mutated. Mutation means, that the elements of DNA are a bit changed. This changes are mainly caused by errors in copying genes from parents. The fitness of an organism is measured by success of the organism in its life.

Search Space:

If we are solving some problem, we are usually looking for some solution, which will be the best among others. The space of all feasible solutions (it means objects among those the desired solution is) is called search space (also state space). Each point in the search space represent one feasible solution. Each feasible solution can be "marked" by its value or fitness for the problem. We are looking for our solution, which is one point (or more) among feasible solutions - that is one point in the search space. The looking for a solution is then equal to a looking for some extreme (minimum or maximum) in the search space. The search space can be whole known by the time of solving a problem, but usually we know only a few points from it and we are generating other points as the process of finding solution continues.

The problem is that the search can be very complicated. One does not know where to look for the solution and where to start. There are many methods, how to find some suitable solution (ie. not necessarily the best solution), for example hill climbing, tabu search, simulated annealing and genetic algorithm. The solution found by this methods is often considered as a good solution, because it is not often possible to prove what is the real optimum.

The basic steps involved in this module are:

i) Encoding
ii) Population Initialization
iii) Fitness Function
iv) Genetic Operators
v) Evaluation and
vi) Termination Criteria

Encoding:

Encoding is the starting point of Genetic Algorithm. Here several types are available like Binary Encoding, Permutation Encoding, Value Encoding and Tree Encoding. In our problem, Binary Encoding has been used, in which Binary value '1' represent presence of an item and '0' represent absence of an item in an itemset. Chromosome length is fixed and it is equal to number of distinct items (n) which is obtained from the transaction database.

Population Initialization:

Given an itemset length 'k', all the genes (item) in a chromosome are encoded as '0'. The initial population is produced using random number generator. If the generated random number is 'r', then the chromosome is encoded as '1' at r_{th} position. This represent i_r item presents in a chromosome (itemset). Upon randomly

generating an item in a chromosome, it is checked against other items already generated in the same chromosome and if the item is present a new number is randomly generated until it is unique. This is repeated until generating 'k' unique random numbers. This process should hold the conditionk n.

Fitness Function:

The main goal this work is to generate the high utility itemsets from the transaction database. Hence, the fitness function is essential for determining the chromosome (itemset) which satisfy minUtil threshold. The following fitness function [15] has been used

$$f(X)=u(X)=\sum_{Tq\in D \wedge X\subseteq Tq} u(X,T_q)$$

where u(X) – utility measure, T- Transaction, D- Database, X-item set.

Genetic Operators:

It would consists of three operators[24] – select, cross over and Mutation. It would select one of the chromosomes, then perform cross over either single point or two point. Mutation serves to prevent premature loss of population by randomly sampling new points in the search space.

Crossover and mutation provide exploration, compared with the exploitation provided by selection. The effectiveness of GA depends on the trade-off between exploitation and exploration.

Crossover: We use one-point crossover in this paper. The crossover operation takes place between two consecutive individuals with probability specified by *crossover rate*. These two individuals exchange portions that are separated by the crossover point The following is an example of crossover:

Indv1: 1 0 1 0 0 1 0 1

Indv2: 1 0 1 1 ↑ 1 1 1 0

Crossover point

After crossover, two offspring are created as below:

Child1: 1 0 1 0 1 1 1 0

Child2: 1 0 1 1 0 1 0 1

Mutation: As discussed earlier, the mutation operator is applied to each bit of an individual with a probability of *mutation rate*. When applied, a bit whose value is 0 is mutated into 1 and vice versa. An example of mutation is as follows.

Indv: 1 1 0 1 1 1 1

↓ ↓

Indv: 1 1 1 0 1 1 0

Selection: The selection process chooses the candidate individuals based on their fitnesses from the population in the current generation. Otherwords, if the chromosome is better, then the chances of getting selected is higher. Some of the selection methods being used are roulette wheel selection, Rank selection, steady-state selection etc. Proportional selection (or roulette wheel selection) is used in this algorithm. It is implemented by using a tournament replacement strategy. Those individuals with higher fitness values are more likely to be selected as the individuals of population in the next generation.

Fitness Evaluation:

The main criteria that we consider here is to evaluate whether the itemset's utility value is greater than the user specified threshold and the fitness function has to be used for that.This method copies the chromosome with higher fitness value to new population.

Termination Criteria:

It is the criterion by which the GA decides whether to continue searching or stop the search. The possible terminating conditions are listed below

1) Fixed number of generations reached

2) The solution's fitness with highest ranking at a fixed number of generations.

3) Manually inspecting the solution

4) Combinations of the above

II. LITERATURE REVIEW

In this section we present a brief overview of the various algorithms, concepts and approaches that have been defined in various research publications.Wide range of studies have been done for mining frequent patterns. Among the issues of frequent pattern mining, the most famous are association rule mining and sequential pattern mining. One of the well-known algorithms for mining association rules is Apriori, which is the pioneer for efficiently mining association rules from large databases. In the paper [13], M.J. Zaki proposed to identify frequent item sets and form the conditional implication rules among them. The algorithms utilize the structural properties of frequent item sets to facilitate fast discovery. The items are organized into subset lattice search space, which is decomposed into small independent chunks or sub-lattices, which can be solved in memory. Efficient lattice traversal techniques are presented, which quickly identify all the long frequent itemsets, and their subsets if required. The main disadvantage is that the performance of algorithm is low.

Agarwal et al in [2] studied the mining of association rules for finding the relationships between data items in large databases . Association rule mining techniques uses a two step process. The first step uses algorithms like the Apriori to identify all the frequent itemsets based on the support value of the itemsets. Apriori uses the downward closure property of itemsets to prune off itemsets which cannot qualify as frequent itemsets by detecting them early. The second step in association rule mining is the generation of association rules from frequent itemsets using the support – confidence model.

Han et al [14] proposed a novel frequent pattern tree (FP-tree) structure, which is an extended prefix-tree structure for storing compressed, crucial information about frequent patterns, and develop an efficient FP-tree based mining method, FP-growth, for mining the complete set of frequent patterns by pattern fragment growth.Efficiency of mining is achieved with three techniques: (i) a large database is compressed into a highly condensed, much smaller data structure, which avoids costly, repeated database scans, (ii) FP-tree-based mining adopts a pattern fragment growth method to avoid the costly generation of a large number of candidate sets, and (iii) a partitioning-based, divide-and-conquer method is used to decompose the mining task into a set of smaller tasks for mining confined patterns in conditional databases, which dramatically reduces the search space. The main limitation is expensive to build and mining from FP-tree.

In the paper [15] proposed by Yun et al, a new algorithm Weighted Interesting Pattern mining (WIP) has been presented, in which a new measure, weight-confidence, is developed to generate weighted hyperclique patterns with similar levels of weights. A weight range is used to decide weight boundaries and an h-confidence serves to identify strong support affinity patterns. WIP not only gives a balance between the two measures of weight and support, but also considers weight affinity and/or support affinity between items within patterns so more valuable patterns can be generated. In the paper [16], Yun proposed weighted frequent pattern mining with length decreasing support constraints. The main approach is to push weight constraints and length decreasing support constraints into the pattern growth algorithm. For pruning techniques, the notion of the Weighted Smallest Valid Extension (WSVE) property with/without Minimum Weight (MinW) has been proposed. The WSVE property with/without MinW is applied to transaction pruning, node pruning and path pruning to eliminate weighted infrequent patterns earlier. Still here the issue of high utility itemset mining is not addressed appropriately.

Liu et al [7] proposed Fast high utility item set mining algorithm, which is mainly composed of two mining phases. In phase I, it employs an Apriori-based level-wise method to enumerate HTWUIs. Candidate itemsets with length k are generated from length k-1 HTWUIs, and their TWUs are computed by scanning the database once in each pass. After the above steps, the complete set of HTWUIs is collected in phase I. In phase II, HTWUIs that are high utility itemsets are identified with an additional database scan. Although two-phase algorithm reduces search space by using TWDC property, it still generates too many candidates to obtain HTWUIs and requires multiple database scans.

In the paper [17], Lin et al addresses the issue related to dynamic environment. In many of the new applications, data flow through the Internet or sensor networks. It is challenging to extend the mining techniques to such a dynamic environment. The main challenges include a quick response to the continuous request, a compact summary of the data stream, and a mechanism that adapts to the limited resources. Distributed programming model for mining business-oriented transactional datasets by using an improved MapReduce framework on Hadoop has been proposed by Vivek et al in the paper [18]. Li et al. [19] proposed an isolated items discarding strategy (IIDS) to reduce the number of candidates. By pruning isolated items during level-wise search, the number of candidate itemsets for HTWUIs in phase I can be reduced. However, this algorithm still scans database for several times and uses a candidate generation-and-test scheme to find high utility itemsets.

In this paper [20], two efficient sliding window-based algorithms, MHUI-BIT (Mining High-Utility Itemsets based on BITvector) and MHUI-TID (Mining High-Utility Itemsets based on TIDlist), are proposed for mining high-utility itemsets from data streams. The advantage is mining high-utility itemsets with negative item profits over stream transaction-sensitive sliding windows but memory issue cannot be overcome as expected. In the paper [21] Tseng et al proposed to discover temporal high utility itemsets which are the itemsets with support larger than a pre-specified threshold in current time window of data stream. A novel approach THUI (Temporal High Utility Itemsets)-Mine has been used for mining temporal high utility itemsets from data streams.

III. PROPOSED WORK

The proposed method can be broadly classified into two stages as mentioned in Fig.1

1. Construct UP tree and identify potentially high utility itemsets (PHUI) using UP growth algorithm.
2. Identify the actual high utility item set from PHUI using genetic algorithm.

In Stage I, the global UP(utility pattern) tree has been constructed with two strategies – DGU (Discarding Global Unpromising Items) and DNU (Decreasing Global Node Utilities). After that, Reorganised Transaction table has been formed with RTUs(Reorganized Transaction Utility).

Construction of Global UP Tree:
In an UP-Tree, each node consists of item name,count, node utility (overestimated utility of node), parent node of N and child node details.A table named header table is employed to facilitate the traversal of UP-Tree. In header table, each entry records an item name, an overestimated utility, and a link. The link points to the last occurrence of the node which has the same item as the entry in the UP-Tree. By following the links in header table and the nodes in UP-Tree, the nodes having the same name can be traversed efficiently.

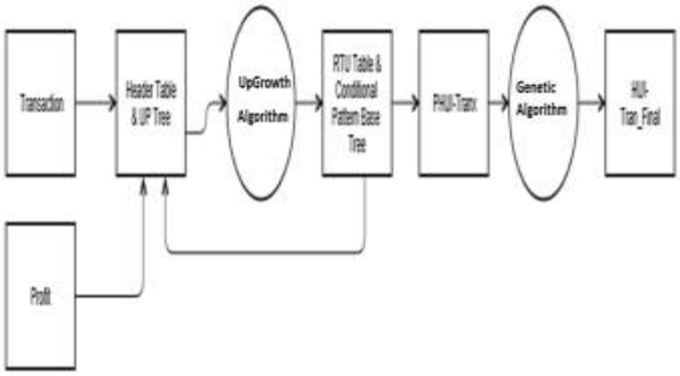

Figure 1: High Utility Itemset mining using UP growth with Genetic Algorithm – Data Flow Diagram

DGU (Discarding Global Unpromising Itemsets):
This involves two scan of database and during the first scan, Transaction Utility (TU) of each transaction is computed as well as TWU of each single item accumulated. In the second scan, transaction are inserted into UP tree and also unpromising items are removed.

DNU (Decreasing Global Node utilities):
This is a Divide and Conquer process that would be useful for large database having lots of transactions. It divide the search space into smaller spaces in such a way that conditional tree has been constructed.

Then UP Growth algorithm started and it involves 2 strategies – DLU (Discarding Local Unpromising items) and DLN (Decreasing Local Node) Utilities. This algorithm is called recursively and generate Potentially High Utility Itemsets (PHUI). The DLU (Discarding Local Unpromising items) and DLN (Discarding Local Node Utilities) is similar to DGU and DNU discussed earlier and has been used to effectively generate PHUI (Potentially High Utility Itemset).

In Stage II, the Genetic algorithm [23] is invoked to mine the actual high utility item sets from the PHUI and optimally generate the required items. The genetic algorithm is chosen because it is a promising solution for global search and it is capable of discovering high utility itemsets with corresponding parameters quantity and profit. Here the basic steps involved are Encoding, Population Intialization, Fitness Function, Genetic Operators, Evaluation and Termination Criteria. To the best of our knowledge, this is the first time with this combination of UP growth and Genetic algorithm is used.

In our case, we are taking distinct items as number of chromosomes and checking it for fitness using the objective function mentioned earlier. We employed tournament replacement strategy for selection of candidate from the population.

Based on the Minimum threshold value, the fitness function has been evaluated and candidate item set is selected for next iteration till the termination criteria reached.

IV. PERFORMANCE COMPARISON

Here we compare the performance of our proposed approach with other algorithm in terms of memory consumed with sample dataset and the approach is better when the threshold value is increasing.

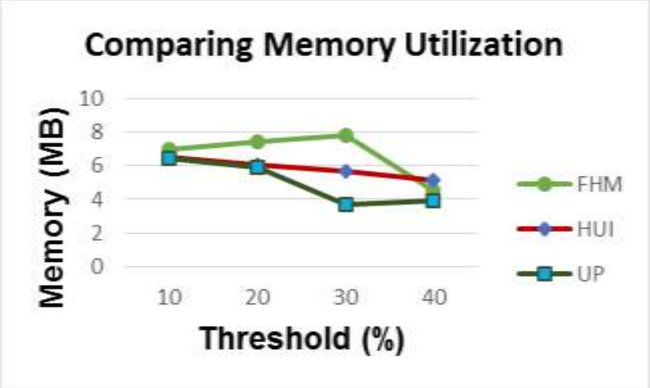

Figure 1: High Utility Itemset mining using UP growth with Genetic Algorithm – Performance Comparison

V. CONCLUSION

Frequent itemset mining is based on the reasoning that the itemsets which appear more frequently in the transaction databases are of more importance to the user .However the practical usefulness of mining the frequent itemset by considering only the frequency of appearance of the itemsets is challenged in many application domains such as Medical research, retail research. It has been that in many real applications that the itemsets that contribute the most in terms of some user defined utility function (for e.g. profit) are not necessarily frequent itemsets.[22]

Hence new concept called Utility mining emerged which attempts to bridge this gap by using item utilities as an suggestive measurement of the importance of that item in the user’s perspective. Here most of the literature work is focused towards reducing the search space while searching for the high utility itemsets.

In this paper we have presented a new approach to use UP growth algorithm with Genetic algorithm. When comparing the performance, the proposed approach shown better results in terms of memory utilization.

REFERENCES

[1] S Laxman , P S Sastry, A survey of temporal data mining , Sadhana , Vol. 31 , part 2 , April 2006, pp. 173-198.

[2] R. Agrawal , T. Imielinski, A. Swami, 1993, mining association rules between sets of items in large databases, in: proceedings of the ACM SIGMOD International Conference on Management of data, pp. 207-216

[3] J.Pillai , O.P.Vyas ,Overview of itemset utility mining and its applications , in: Internationa Journal of Computer Applications (0975-8887), Volume 5-No.11(August 2010)

[4] R. Agrawal, R Srikant, Fast algorithms for mining association rules,in : Proceedings of 20th international Conference on Very Large Databases ,Santiago, Chile, 1994, pp.487-499

[5] H.Mannila , H.Toivonen, Levelwise search and borders of theories in knowledge discovery, Data Mining and Knowedge Discovery 1(3)(1997) 241-258

[6] C.Silverstein, S. Brin , R. Motwani, Beyond market basket: generalizing association rules to dependence rules, Data Mining and Knowledge Discovery 2(1) (1998) pp.39-68

[7] Liu. Y, Liao. W,A. Choudhary, A fast high utility itemsets mining algorithm, in: Proceedings of the Utility-Based Data Mining Workshp, August 2005

[8] H.Yao,H.J.Hamilton ,Mining itemset utilities from transacation databases, in Data and Knowledge Enineering 59(2006) pp.603-626

[9] L.Szathmary,A.Napoli, P.Valtchev, Towards rare itemset mining ,in: Proceedings of the 19th IEEE Interational Conference on Tools with Artificial Intelligence , 2007, Volume-1, pp.305-312,

[10] Guangzhou Yu, Shihuang Shao and Xianhui Zeng mining long high utility itemsets in transaction databases wseas transactions on information science & applications issue 2, volume 5, feb. 2008

[11] J Pei,J. Han, L.V.S. Lakshmanan, Pushing convertible constraints in frequent itemset mining, Data Mining and Knowledge Discovery 8(3)(2004) 227-252

[12] H.Yao, H.J.Hamilton, C.J.Butz, A foundation approach to mining itemset utilities from databases,in:Proceedings of the Third SIAM International Conference on Data Mining, Orlando, Florida , 2004,pp.482-486

[13] M.J. Zaki, "Scalable Algorithms for Association Mining", IEEE Trans. Knowledge and Data Eng., vol. 12, no. 3, pp. 372-390, May 2000.

[14] J. Han, J. Pei, and Y. Yin, "Mining Frequent Patterns without Candidate Generation", Proc. ACM-SIGMOD Int’l Conf. Management of Data, pp. 1-12, 2000.

[15] U. Yun and J.J. Leggett, "WIP: Mining Weighted Interesting Patterns with a Strong Weight and/or Support Affinity", Proc. SIAM Int’l Conf. Data Mining (SDM ’06), pp. 623-627, Apr. 2006.

[16] U. Yun, "An Efficient Mining of Weighted Frequent Patterns with Length Decreasing Support Constraints", Knowledge-Based Systems, vol. 21, no. 8, pp. 741-752, Dec. 2008.

[17] C.H. Lin, D.Y. Chiu, Y.H. Wu, and A.L.P. Chen, "Mining Frequent Itemsets from Data Streams with a Time-Sensitive Sliding Window", Proc. SIAM Int’l Conf. Data Mining (SDM ’05), 2005.

[18] Arati W. Borkar, Dr. Sanjay T. Singh, "Improved Map reduce Framework using High Utility Transactional Databases", International Journal of Engineering Inventions,Volume 3, Issue 12 (July 2014) PP: 49-55

[19] Y.-C. Li, J.-S. Yeh, and C.-C. Chang, "Isolated Items Discarding Strategy for Discovering High Utility Itemsets," Data and Knowledge Eng., vol. 64, no. 1, pp. 198-217, Jan. 2008.

[20] H.F. Li, H.Y. Huang, Y.C. Chen, Y.J. Liu, and S.Y. Lee, "Fast and Memory Efficient Mining of High Utility Itemsets in Data Streams", Proc. IEEE Eighth Int’l Conf. on Data Mining, pp. 881-886, 2008.

[21] V.S. Tseng, C.J. Chu, and T. Liang, "Efficient Mining of Temporal High Utility Itemsets from Data Streams", Proc. ACM KDD Workshop Utility-Based Data Mining Workshop (UBDM ’06), Aug.2006.

[22] Sudip Bhattacharya, Deepty Dubey, “High Utility Itemset Mining”, International Journal of Emerging Technology and Advanced Engineering, Volume 2, Issue 8, August 2012

[23] S. Kannimuthu, Dr. K.Premalatha, "Discovery of High Utility Itemsets Using Genetic Algorithm", International Journal of Engineering and Technology (IJET),Vol 5 No 6,pp.4866-4880 Dec 2013-Jan 2014

[24] Venkatesh S,K.M.Mehata, "A Fault tolerant system based on Genetic Algorithm for Target Tracking in Wireless Sensor Networks", International Journal of Computer Applications Technology and Research Volume 3– Issue 7, 434 - 438, 2014

[25]M.Mahalakshmi, P. Kalaivani, E.Kiruba Nesamalar, "Review on Genetic Algorithm and its Applications",International Journal of Computing Algorithm,Volume: 02, December 2013, Pages: 415-423