



# Discovering Interesting Association Rules: A Multi-objective Genetic Algorithm Approach

Basheer Mohamad Al-Maqaleh  
Faculty of Computer Sciences &  
Information Systems, Tamar  
University, Yemen

## ABSTRACT

Association rule mining is considered as one of the important tasks of data mining intended towards decision making process. It has been mainly developed to identify interesting associations and/or correlation relationships between frequent itemsets in datasets. A multi-objective genetic algorithm approach is proposed in this paper for the discovery of interesting association rules with multiple criteria i.e. support, confidence and simplicity (comprehensibility). With Genetic Algorithm (GA), a global search can be achieved and system automation is developed, because the proposed algorithm could identify interesting association rules from a dataset without having the user-specified thresholds of minimum support and minimum confidence. The experimental results on various types of datasets show the usefulness and effectiveness of the proposed algorithm.

## General Terms

Knowledge Discovery in Databases (KDD), Data Mining, Genetic Algorithm, Optimization.

## Keywords

Association Rule, Interestingness Measure, Genetic Operators, Frequent Itemsets.

## 1. INTRODUCTION

Knowledge Discovery in Databases (KDD) has been an active and attractive research challenge both in the areas of artificial intelligence and databases. KDD is aimed at developing methodologies and tools which can automate the data analysis process and create useful information and knowledge from data to help in decision making [1], [2]. A widely accepted definition is given by Fayyad et al. [3] in which KDD is defined as the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data. This definition points to KDD as a complicated process comprising a number of steps. Data mining is one step in this process. Data mining is an essential process, where intelligent methods are applied in order to extract data patterns [4]. One of the important areas in data mining is association rule mining. Since its introduction in 1993 [5] the area of association rule mining has received a great deal of attention. Association rules are an intuitive descriptive paradigm that has been used extensively in different application domains with the purpose to identify the regularities and correlation in a set of observed objects [4]. It is an implication of the form  $A \rightarrow B$ , where A (antecedent part) and B (consequent part) are frequent itemsets in a transaction database and  $A \cap B = \emptyset$ . The rule  $A \rightarrow B$  can be interpreted as “if itemset A occurs in a transaction, then itemset B will also likely occur in the same transaction”. For example, suppose in a database 27% of all

transactions contain both bread and milk and 90% of all transactions contain bread. An association rule mining system might therefore derive the rule bread  $\rightarrow$  milk with 27% support and 90% confidence. Rule support and confidence are two measures of rule interestingness. They, respectively, reflect the usefulness and certainty of the discovered rules [4]. A support of 27% means that 27% of all the transactions under analysis show that bread and milk are purchased together. A confidence of 90% means that 90% of the customers who purchased bread also bought the milk. Typically, association rules are considered interesting if they satisfy both a minimum support threshold and a minimum confidence threshold [4]. Such thresholds can be set by users or domain experts [2]. The association rule discovery task consists of extracting from the data being mined all rules with support and confidence greater than or equal to user-specified thresholds [4]. Association rules enable us to detect the items that frequently occur together in an application [6]. Genetic Algorithm (GAs) as are computer based optimization methods based on biological mechanisms, such as, Mendel’s laws and Darwin’s fundamental principle of natural selection. It imitates the mechanics of natural species evolution with genetics principles, such as natural selection, crossover and mutation [7]. A GA searches for good solutions to a problem by maintaining a population of candidate solutions and creating subsequent generations by selecting the current best solutions and using operators like crossover and mutation to create new candidate solutions. Thus, better and better solutions are “evolved” over time. Commonly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population [7], [8], [9]. The advantage of GA becomes clearer when the search space of a task is large [10]. There have been many applications of GAs in the field of data mining and knowledge discovery. Most of them are addressed to the problem of classification [11], [12], [13], [14], [15], [16], [17], [18], [19], [20]. The GAs are important when discovering association rules as the rules that GA found are usually more general because of its global search nature to discover the set of items frequency and they are less complex than other induction algorithms often used in data mining, where these algorithms usually performs a kind of local search [21]. As a result of their global search, GAs tend to cope better with attribute interactions than inductions algorithms [10], [15]. Han et al. [22] designed a strategy to mine top-k frequent closed patterns for effectiveness and efficiency. Cheung and Fu [23] developed a technique to identify frequent itemsets without the support threshold. Yan et al. [21] designed a GA-based strategy for identifying association rules without specifying actual minimum support. In their approach, an elaborate encoding method is developed,



and the relative confidence is used as the fitness function. A novel association rules method based on GA and fuzzy set strategy for web mining is described in [24]. It is based on a hybrid technique that combines the strengths of rough set theory and GA. Multi-objective rule mining using GAs is presented in [25]. In their work they used the comprehensibility and the interestingness measure of the rules in addition to predictive accuracy. Mata et al. [26] use a GA to optimize the support of an interval for a quantitative attribute. However, their approach is limited to datasets without any overlap among different classes of the data.

In this paper, the multi-objective genetic algorithm approach is used to discover interesting association rules from large datasets. The use of a minimum support and minimum confidence threshold results in the loss of interesting association rules if these thresholds are set inaccurately. So, the proposed algorithm does not require users to specify the minimum-support and minimum confidence thresholds. Instead of generating a large number of rules in traditional data mining approaches, only the most interesting rules are discovered according to the interestingness measure (objective) defined by the fitness function. This approach enables the user to discover knowledge from any transactional record without the background knowledge of an application domain usually necessary to establish a threshold prior to mining. The GA in this paper is proposed for the discovery of interesting association rules with multiple criteria i.e. support, confidence and simplicity.

The remainder of this paper is organized as follows. In Section 2, the proposed GA-based model for identifying association rules, including the encoding method, genetic operators, and fitness function are presented. Section 3, reports the computational results. Finally, Section 4 presents the conclusions and discusses future work.

## 2. THE PROPOSED APPROACH

In this section a multi-objective genetic algorithm model is presented for the automated discovery of interesting association rules from large datasets. This section discusses the encoding, genetic operators, and fitness function used in the proposed approach as given below.

### 2.1 Encoding

To apply a GA, first a suitable encoding needs to be chosen to represent candidate solutions to the given problem. Representation of rules plays a major role in GAs, broadly there are two approaches based on how rules are encoded in the population of individuals (chromosomes) [10]. One encoding method is called Michigan approach, where each rule is encoded into an individual. Another is referred to as Pittsburgh approach, in which a set of rules are encoded into a chromosome. In this paper Michigan’s approach is opted i.e. each individual encodes a single rule. The structure of an individual is made up of genes and is represented as follows:

Suppose there are  $n$  predicting attributes in the data being mined. An individual (sequence of genes) corresponds to a single association rule is divided into two parts: antecedent part consisting of a conjunction of conditions on the values of the predicting attributes, and consequent part consisting of conjunction of conditions on the values of predicting attributes. Accordingly, we have fixed length  $(2n)$ , individual representation. It is to be noted that any of the  $n$  attributes (one or more) can form conditions in the antecedent part and

similarly any of the  $n$  attributes (one or more) can form conditions in the antecedent part. Further, for any rule the set of attributes forming antecedent part and the set of attributes forming consequent part would be disjoint, i.e. (set of attributes present in the antecedent part)  $\cap$  (set of attributes present in the consequent part) =  $\emptyset$ . The genes are positional, i.e. the first gene represents the first attribute, the second gene represents the second attribute and so on. If an attribute is not present in the rule the corresponding value in gene is “#”. The structure of individual is shown in Fig. 1.

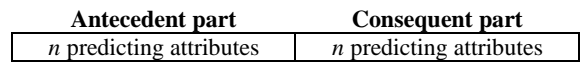


Fig. 1: The structure of individual

For example, consider the description of the Contact-lenses dataset [27] given in Table 1.

Table 1. Description of the Contact-Lenses dataset.

Attribute	Possible values	Allies
Age	young, pre-presbyopic, presbyopic	'1', '2', '3'
Prescription(Pre.)	myope, hypermetrope	'1', '2'
Astigmatism(Ast.)	no, yes	'1', '2'
Rate	reduced, normal	'1', '2'
Lenses(Len.)	none, soft, hard	'1', '2'

Corresponding to the above dataset, an association rule: Ast. = yes  $\wedge$  Len. =none  $\rightarrow$  Rate=reduced, would be encoded as:

Age	Pre.	Ast.	Rate	Len.	Age	Pre.	Ast.	Rate	Len.
#	#	2	#	1	#	#	#	1	#

### 2.2 Genetic Operators

Genetic operators are some of the most important components of GAs. The conventional genetic operators for selection, crossover and mutation are used. More precisely, in the proposed system fitness proportional selection, one-point crossover with crossover probability of 95% and mutation operator with probability of 10% for each kind of mutation are used. An elitist reproduction strategy is also used, where the best individual of each generation was passed unaltered to the next generation. The crossover operator takes two randomly chosen parent individuals as input, combines them and generates two children. This process of combining is carried out by, randomly, choosing a crossover point in the strings of the parents and then swapping the genes to the right of the crossover point between the two parent individuals, yielding the new child individuals. The mutation operator is an operator that acts on a single individual at a time. It alters a gene of a chromosome and, thus, brings variability into the population of the subsequent generation. It is used to ensure that all possible alleles can enter the population and therefore maintain the population diversity [28]. In the proposed system the mutation does the following operations [14]:

1. Insertion: When a hash (#) is mutated to a non-hash value, it works like an insertion operator. While applying the mutation operator, insert to the antecedent part or consequent part, it must be ensured that the mutated chromosome is legal i.e., the condition (set of attributes present in the antecedent



part)  $\cap$  (set of attributes present in the consequent part) =  $\emptyset$  is satisfied. In case the above condition is not satisfied, the produced mutated chromosome is rejected as illegal and the operator insert is applied again to produce legal mutant.

2. Deletion: When a non-hash value is mutated to a hash, it works like deletion operator.

3. Value mutation: Value of an attribute can be mutated to another value of the same attribute.

### 2.3 Fitness Function

Fitness functions are used to measure the quality of rule. It is very important to define a good fitness function that rewards the right kinds of individuals. The fitness function is always problem dependent. Multi objective processing can be fostered for mining the interesting association rules. Based on that, in the present work three important measures of the rules like support, confidence and simplicity are considered. Using these three measures, some previously unknown, easily understandable and compact rules can be generated. So, association rule mining problems can be considered as a multi-objective problem rather than as a single objective one [25]. The support (sup) of the association rule  $A \rightarrow B$  is defined as:

$$\text{sup} = \frac{|A \cap B|}{|N|} \quad (1)$$

where,  $|N|$  is the total number of transactions, and  $|A \cap B|$  is the number of transactions containing both A and B. Support is often used to eliminate non interesting rules and can be considered as an indication of how often a rule occurs in a dataset [29]. A measure to predict the association rule precision  $A \rightarrow B$  is the confidence (conf). This measures the reliability of inference made by the rule which is defined like:

$$\text{conf} = \frac{|A \cap B|}{|A|} \quad (2)$$

where  $|A|$  is the number of transactions containing A. A higher confidence suggests a strong association between A and B. But the confidence favours the rules overfitting the data [23]. The generated rule may have a large number of attributes involved in the rule, thereby making it difficult to understand. If the discovered rules are not simple and understandable to the user, the user will never use them. So the simplicity (simp) measure is needed to make the discovered rules are easy to understand. The simplicity of the rule can be defined by the number of attributes on the left hand side of the rule (antecedent part) and tries to quantify the understandability of the rule. This expression serves well for the classification rule generation where the number of attributes in the consequent part is always one. Since, in the association rules, the consequent part may contain more than one attribute; this expression is not suitable for the association rule mining [30]. Where the number of attributes involved in both the parts of the rule has some effect an expression is required. Simplicity of an association rule is quantified by the following expression [25]:

$$\text{simp} = \frac{\log(1 + |B|)}{\log(1 + |A \cup B|)} \quad (3)$$

where  $|B|$  and  $|A \cup B|$  are the number of attributes involved in the consequent part and the total rule, respectively. This measure helps to generate simpler and more concise association rules. The rule is considered more simple, if the

number of conditions in the antecedent part is less. As mentioned above, association rule mining is viewed as a multi-objective problem rather than single objective one. So, the fitness function is defined as below:

$$\text{fitness} = \frac{(w1 \cdot \text{sup} + w2 \cdot \text{conf} + w3 \cdot \text{simp})}{(w1 + w2 + w3)} \quad (4)$$

where  $w1$ ,  $w2$  and  $w3$  are user-defined weights. Since finding the frequent itemsets is of great computational complexity, the problem of mining association rules can be reduced to the problem of finding frequent itemsets[26]. Based on this, in the proposed system the weight values of  $w1=3$ ,  $w2=2$  and  $w3=1$  were chosen according to the proposed evaluation about relative importance of support, confidence and simplicity. Note that the value of fitness is normalized in the range of [0..1].

### 3. COMPUTATIONAL RESULTS

The performance of the suggested approach is validated on four sets of data obtained from UCI Machine Learning Repository, which is a collection of widely used benchmark and real-world datasets for data mining and KDD community [27]. For each of the dataset, the proposed GA had 100 individuals in the population and was run for 500 generations. The proposed algorithm was terminated when the maximum number of generations has reached. The performance of proposed algorithm is evaluated and compared with the well-known Apriori algorithm which is implemented in a public domain tool called Weka[31]. The default parameters of the Apriori algorithm are used to make the comparison as fair as possible. Also, the default parameters of the proposed GA are used without any attempt to optimize parameters. When comparing the proposed algorithm with the Apriori algorithm, the value of the minimum support and minimum confidence are provided by the user to the Apriori algorithm. Nevertheless, these values can be automatically obtained by looking for the lowest value of sup column and conf column in the results of the proposed algorithm. The results reported below for those four datasets are an average over the 5 runs. The performance of the proposed algorithm on different datasets is demonstrated below:

#### 3.1 Experiments

##### 3.1.1 Experiment One

The Led 7 dataset was used for this experiment. This dataset contains 3200 instances and 8 attributes. Table 2 shows the interesting association rules generated from this dataset. In Table 2 the lowest value in the sup column is 0.197 and the lowest value in the conf column is 0.900. So, these two values are used as the threshold values of the minimum support and minimum confidence in the Apriori algorithm respectively. Based on these constraints, the Apriori algorithm would generate 14 association rules from the Led 7 dataset.

##### 3.1.2 Experiment Two

This experiment was carried out on the Tic-Tac-Toe dataset (958 instances and 10 attributes). The proposed algorithm generated the following interesting association rules as shown in Table 3. The Apriori algorithm with minimum support=0.200 and minimum confidence=0.560 would generate 16 association rules from the same dataset.



**Table 2. The result for the Led 7 dataset.**

No	Discovered Rules	sup	conf	simp	fitness
1	Attr#4=1 $\wedge$ Attr#6=1 $\wedge$ Attr#7= 1 $\rightarrow$ Attr#1 =1	0.352	0.920	0.431	0.555
2	Attr#5=1 $\wedge$ Attr#7=1 $\rightarrow$ Attr#1=1	0.331	0.910	0.500	0.552
3	Attr#2=1 $\wedge$ Attr#5=0 $\rightarrow$ Attr#6=1	0.267	0.900	0.500	0.517
4	Attr#5=0 $\wedge$ Attr#7=0 $\rightarrow$ Attr#6=1	0.252	0.900	0.500	0.509
5	Attr#4=1 $\wedge$ Attr#5=1 $\wedge$ Attr#7= 1 $\rightarrow$ Attr#1 =1	0.239	0.910	0.431	0.495
6	Attr#3=0 $\wedge$ Attr#7=1 $\rightarrow$ Attr#1 =1	0.197	0.920	0.500	0.489
7	Attr#1=1 $\wedge$ Attr#2=1 $\wedge$ Attr#5= 1 $\rightarrow$ Attr#7 =1	0.200	0.910	0.431	0.492

**Table 3. The result for the Tic-Tac-Toe dataset.**

No	Discovered Rules	sup	conf	simp	fitness
1	t5=x $\rightarrow$ class=positive	0.382	0.800	0.631	0.563
2	t9=x $\rightarrow$ class=positive	0.308	0.710	0.631	0.496
3	t8=o $\rightarrow$ class=positive	0.239	0.690	0.631	0.455
4	class=positive $\rightarrow$ t5=x	0.382	0.580	0.631	0.490
5	class=negative $\rightarrow$ t5=o	0.200	0.560	0.631	0.392

### 3.1.3 Experiment Three

This experiment was carried out on the Balance Scale dataset. This dataset contains 625 instances and 5 attributes. Table 4 shows the discovered interesting association rules by the proposed algorithm. Form this dataset with minimum support=0.040 and minimum confidence=0.700, the Apriori algorithm would generate 8 association rules.

### 3.1.4 Experiment Four

This experiment was carried out on the Nursery (12960 instances and 9 attributes). Table 5 shows the interesting association rules generated from this dataset. With minimum support=0.111 and minimum confidence=1.000, the Apriori algorithm would generate 24 association rules form the same dataset.

## 3.2 Summary of the results and discussion

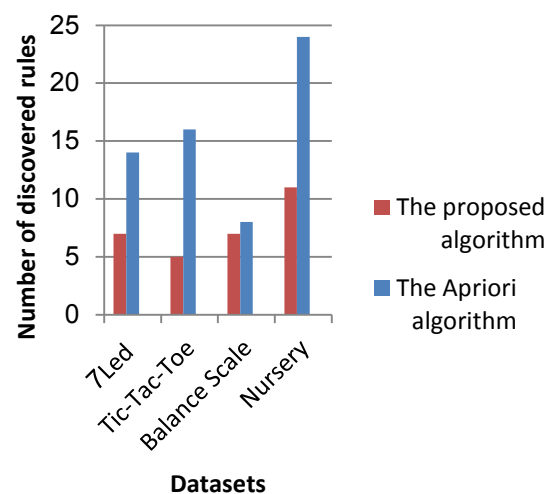
The largest number of discovered association rules is a practical bottleneck for algorithm performance. If the number of discovered rules is large, the interpretation becomes difficult [32]. Fig. 2 depicts the comparative performance of the two approaches in term of number of discovered rules. From this figure it can be observed that the proposed algorithm generates less number of association rules than the Apriori algorithm for the four datasets used in the experiments. The large number of rules generated by the Apriori algorithm makes manual inspection of the rules very difficult. It is hence impossible for an expert of the field being mined to sustain these rules.

**Table 4. The result for the Balance Scale dataset.**

No	Discovered Rules	sup	conf	simp	fitness
1	left-weight=1 $\rightarrow$ class = Right	0.160	0.800	0.630	0.452
2	left-distance=1 $\rightarrow$ class= Right	0.160	0.780	0.630	0.444
3	left-weight=1 $\wedge$ left-distance=1 $\rightarrow$ class=Right	0.040	0.960	0.500	0.423
4	left-weight=1 $\wedge$ right-weight=5 $\rightarrow$ class=Right	0.040	0.960	0.500	0.423
5	left-weight=5 $\wedge$ left-distance=5 $\rightarrow$ class=Left	0.040	0.960	0.500	0.423
6	left-weight=5 $\wedge$ right-weight =1 $\rightarrow$ class=Left	0.040	0.960	0.500	0.423
7	left-distance=5 $\rightarrow$ class= Left	0.140	0.700	0.630	0.408

This is particularly true for datasets whose attributes are highly correlated. Overall, the computational results indicate that the proposed GA considerably outperformed the Apriori algorithm in those four datasets, with respect to the number of discovered rules.

Table 6 shows the average performance of the two algorithms. The values of sup and conf in this table refer to the total average of support and confidence for the generated rules, respectively, using the specified algorithm for the dataset under consideration. Fig. 3 and Fig. 4 depict the comparative performance of the two algorithms based on the average of support and average of confidence respectively. If we look at the average performance of the two algorithms, the proposed algorithm performs better than the Apriori algorithm. Except in Balance Scale dataset, the average of support in the Apriori algorithm is better than the proposed algorithm. Since it is a multi-objective problem, we cannot prioritize one objective over another.



**Fig. 2: Number of discovered rules by the proposed algorithm and the Apriori algorithm.**

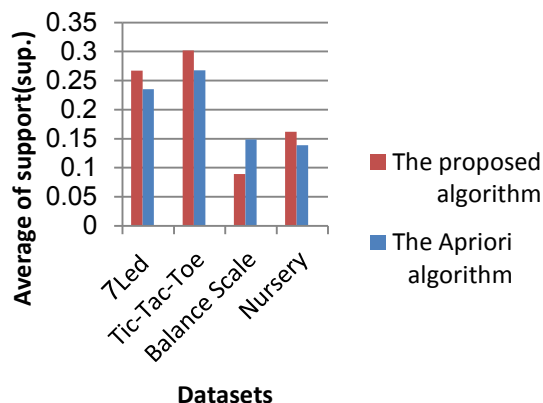


**Table 5. The result for the Nursery dataset.**

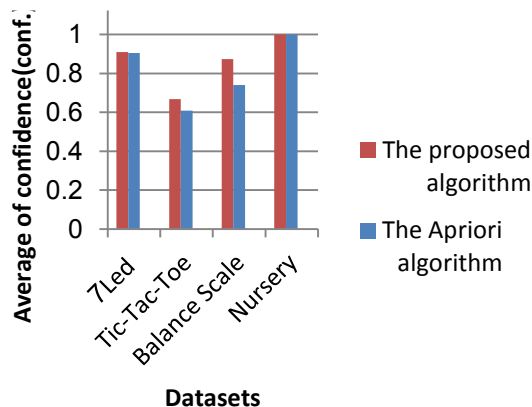
No.	Discovered Rules	sup	conf	simp	fitness
1	class=not_recom → health=not_recom.	0.333	1.000	0.631	0.605
2	health=not_recom→ class=not_recom	0.333	1.000	0.631	0.605
3	finance=convenient^class=not_recm →health= not_recom	0.167	1.000	0.500	0.500
4	finance=inconvenient ^ health=not_recom → class=not_recm	0.167	1.000	0.500	0.500
5	parents= usual ^ class=not_recm → health= not_recom	0.111	1.000	0.500	0.473
6	parents=pretentious ^ class=not_recm → health= not_recom	0.111	1.000	0.500	0.473
7	housing= less_con ^class=not_recm → health= not_recom	0.111	1.000	0.500	0.473
8	housing= critical ^ health=not_recom → class=not_recm	0.111	1.000	0.500	0.473
9	social=non_prob ^ health=not_recom → class=not_recm	0.111	1.000	0.500	0.473
10	social=slightly_prob ^ class=not_recm → health=not_recom	0.111	1.000	0.500	0.473
11	social=problematic ^ health=not_recom → class=not_recm	0.111	1.000	0.500	0.473

**Table 6. Average performance of two algorithms.**

Data set	Algorithms	sup	conf
Led 7	The proposed algorithm	0.267	0.910
	The Apriori algorithm	0.235	0.904
Tic-Tac-Toe	The proposed algorithm	0.302	0.668
	The Apriori algorithm	0.268	0.608
Balance Scale	The proposed algorithm	0.089	0.874
	The Apriori algorithm	0.149	0.741
Nursery	The proposed algorithm	0.162	1.000
	The Apriori algorithm	0.139	1.000



**Fig. 3: Average of support of discovered rules by the proposed algorithm and the Apriori algorithm.**



**Fig. 4: Average of confidence of discovered rules by the proposed algorithm and the Apriori algorithm.**

#### 4. CONCLUSION AND FUTURE WORK

Association rule mining is an important topic in data mining and receiving increasing attention. The multi-objective genetic algorithm approach for the discovery of interesting association rules is proposed in this paper. The proposed algorithm is alternative to find a set of interesting association rules with multiple criteria i.e. support, confidence and simplicity. The most important difference between the proposed algorithm and the existing mining strategies is that the proposed algorithm does not require the minimum support and minimum confidence thresholds. The results show that the proposed model can attain considerable performance improvement in terms of the interesting association rules discovery and the number of discovered rules comparing to the Apriori algorithm. One of the most important future research directions would be the discovery of weighted fuzzy association rules from large datasets using the evolutionary algorithm.

#### 5. REFERENCES

- [1] Bramer, M. 2007. Principles of Data Mining. Springer-Verlag London Limited.
- [2] Tan, P-N, Steinbach, M. and Kumar V. 2006. Introduction to Data Mining. Addison-Wesley.
- [3] Fayyad, U. and Uthurusamy, R. 1996 .Data mining and knowledge discovery in databases. Communications of the ACM, vol. 39, no. 11, pp.24–34.
- [4] Han, J. and Kamber, M. 2006. Data Mining Concepts and Techniques. Morgan Kaufmann.
- [5] Agrawal, R., Imielinski, T. and Swami, T. 1993. Mining association rules between sets of items in large databases. In Proceedings of ACM SIGMOD International Conference on Management of Data (SIGMOD’ 93), pp. 207–216.
- [6] Koh, Y. S. and Rountree, N. 2009. Rare association rule mining: An overview. In Yun, S. K. and Nathan R. (Eds.), Rare Association Rule Mining and Knowledge Discovery: Technologies for Infrequent and Critical Event Detection, ISBN: 978-1-60566-754-6, Information Science Reference. pp. 1-14.
- [7] Goldberg, D. E. 1989. Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley.



- [8] Sivanandam, S. N. and Deepa, S. N. 2008. Introduction to Genetic Algorithms. Springer-Verlag Berlin Heidelberg.
- [9] Bharadwaj, K. K., Hewahi, N. M. and Brando, M. A. 1996. Adaptive hierarchical censored production rule-based system: A genetic algorithm approach. *Advances in Artificial Intelligence, SBIA '96, Lecture Notes in Artificial Intelligence*, no. 1159, Berlin, Germany, Springer-Verlag, pp. 81-90.
- [10] Frietas, A. A. 2002. Data Mining and Knowledge Discovery with Evolutionary Algorithms. Springer-Verlag Berlin Heidelberg.
- [11] Al-Maqaleh, B. M. 2012. Genetic algorithm approach to automated discovery of comprehensible production rules. In *Proceedings of the 2nd International Conference on Advanced Computing & Communication Technologies (ACCT2012)*, IEEE Computer Society, Rohtak, India, pp. 69-71.
- [12] Carvalho, D. R. and Frietas, A. A. 2002. A genetic algorithm for discovering small-disjunct rules in data mining. *Applied Soft Computing*, vol.2, no. 1, pp.75-88.
- [13] Sarkar, B. K., Sana, S. S. and Chaudhuri, K. 2012. A genetic algorithm-based rule extraction system. *Applied Soft Computing*, vol.12, pp. 238-254.
- [14] Saroj, and Bharadwaj, K. K. 2009. Discovery of exceptions: A step towards perfection. In *Proceedings of the 3rd International Conference on Network and System Security*, IEEE Computer Society, pp. 540-545.
- [15] Dehuri, S. and Mall, R. 2006. Predictive and comprehensible rule discovery using a multi objective genetic algorithm. *Knowledge Based Systems*, vol. 19, pp. 413-421.
- [16] Al-Maqaleh, B. M. and Bharadwaj, K. K. 2007. Evolutionary approach to automated discovery of censored production rules with fuzzy hierarchy. In *Proceedings of the International Conference on Data Mining and Applications (ICDMA'2007)*, Hong Kong, China, vol. 1, pp. 716-721.
- [17] Au, W. and Chan, K. 2002. An evolutionary approach for discovering changing patterns in historical data. In *Proceedings of 2002 SPIE*, vol. 4730, pp. 398-409.
- [18] Noda, E., Freitas, A. A. and Lopes, H. S. 1999. Discovering interesting prediction rules with a genetic algorithm. In *Proceedings of 1999 Congress on Evolutionary Computation (CEC' 99)*, pp. 1322-1329.
- [19] Al-Maqaleh, B., Al-Dhobai, M. and Shabazakia, H. 2012. An evolutionary algorithm for automated discovery of small-disjunct rules. *International Journal of Computer Applications*, vol. 41, issue 8, pp.33-37.
- [20] Al-Maqaleh, B. and Shabazakia, H. 2012. A genetic algorithm for discovering classification rules in data mining. *International Journal of Computer Applications*, vol.41, issue 18, pp.40- 44.
- [21] Yan, X., Zhang, C. and Zhang, S. 2009. Genetic algorithm- based strategy for identifying association rules without specifying actual minimum support. *Expert Systems with Applications*, vol. 36, pp. 3066-3076.
- [22] Han, J., Wang, J., Lu, Y. and Tzvetkov, P. 2002. Mining top-k frequent closed patterns without minimum support. In *Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM 2002)*, pp. 211-218.
- [23] Cheung, Y. and Fu, A. 2004. Mining frequent itemsets without support threshold: With and without item constraints. *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, pp. 1052-1069.
- [24] Chai, C. and Li, B. 2010. A novel association rules method based on genetic algorithm and fuzzy set strategy for web mining. *Journal of Computers*, vol. 5, no. 9, pp. 1448-1455.
- [25] Ghosh, A. and Nath, B. 2004. Multi-objective rule mining using genetic algorithms. *Information Sciences*, vol. 163, pp. 123-133.
- [26] Mata, J., Alvarez, J. L. and Riquelme, J. C. 2002. An evolutionary algorithm to discover numeric association rules. In *Proceedings of ACM SAC Symp. Applied Computing*, Madrid, Spain, pp. 590-594.
- [27] UCI Repository of Machine Learning Databases, Department of Information and Computer Science University of California, 1994. [<http://www.ics.uci.edu/~mllearn/MLRepository.html>].
- [28] Sewell, M., Samarabandu, J., Rodrigo, R. and McIsaac, K. 2006. The rank-scaled mutation rate for genetic algorithms. *International Journal of Information Technology*, vol. 3, no. 1.
- [29] Webb, G. 2000. Efficient search for association rules. In *Proceedings of International Conference on Knowledge Discovery and Data Mining*, pp. 99-107.
- [30] Liu, B., Hsu, W. and Ma, Y. 1998. Integrating classification and association rule mining. In *Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining (KDD-98)*, AAAI.
- [31] Weka:<http://www.cs.waikato.ac.nz/ml/weka/index.html>.
- [32] Aldosari, B., Almodaifer, G., Hafez, A. and Mathkour, H. 2012. Constrained association rules for medical data. *Journal of Applied Sciences, Asian Networks for Scientific Information*, vol. 12, issue 17, pp. 1792-1800.