A Multi-Objectives Genetic Algorithm Clustering Ensembles Based Approach to Summarize Relational Data

Rayner Alfred[1], Gabriel Jong Chiye[1], Yuto Lim[2], Chin Kim On[1] and Joe Henry Obit[1]

[1]Faculty of Computing and Informatics, Universiti Malaysia Sabah, Malaysia
[2]School of Information Science, Japan Advanced Institute of Science and Technology, Japan
ralfred@ums.edu.my, gabrieljongums@gmail.com, ylim@jaist.ac.jp, kimonchin@ums.edu.my, joehenry@ums.edu.my

Abstract. Dynamic Aggregation of Relational Attributes is one of the approaches which can be used to learn relational data. It is capable to transform a multi-relational database into a vector space representation. Traditional clustering algorithm can then be applied directly on the vector space representation to learn and summarize the relational data. However, the performance of the algorithm is highly dependent on the quality of clusters produced. A small change in the initialization of the clustering algorithm parameters may cause adverse effects to the clusters quality produced. In order to optimize the quality of clusters, a Genetic Algorithm is used to find the best combination of initializations and settings to produce the optimal clusters. The proposed method involves the task of finding the best initialization with respect to the number of clusters, proximity distance measurements, fitness functions, and classifiers used for the evaluation. Based on the results obtained, clustering coupled with Euclidean distance is found to perform better in the classification stage compared to using clustering coupled with Cosine similarity. Based on the findings, the cluster entropy is the best fitness function, followed by multi-objectives fitness function used in the genetic algorithm. This is most probably because of the involvement of external measurement that takes the class label into consideration in optimizing the structure of the cluster results.

Keywords: Relational Data Mining, *k*-Means, Clustering, Ensembles, Genetic Algorithm, Multi-Objectives.

1 Introduction

Data mining is the process of discovering interesting patterns and knowledge from large amounts of data [1][2]. It involves several stages, first is the data preprocessing, which relevant data is selected and retrieved from the databases, cleaning to remove noise and handle missing data, integration to combine data from multiple sources. Second, data transformation, where features selection and transformation is applied, to produce an appropriate representation which represent the databases. Third, data mining process in which intelligent methods are used to extract patterns, such as rules,

clusters and etc. Finally, pattern evaluation and presentation, where interesting patterns extracted are presented in knowledge which is easily human understandable.

From database point of view, a typical relational database consists of multiple relations known as tables [3][4]. These relations are connected via semantic links such as entity relationship links. Many useful traditional data mining algorithms work on single table form only and cannot be applied directly in learning multi-relational databases [5]. Flattening the multi-relational databases leads will lead to inaccurate clustering and results in wrong business decision.

Thus, several approaches have been proposed to learn multi-relational databases. Inductive Logic Programming [6], Propositionalization [7] and Dynamic Aggregation for Relational Attributes [8] are popular approaches in learning relational datasets. Dynamic Aggregation of Relational Attributes is designed based on a clustering technique [9][10] in which it is capable of transforming multi-relational databases to vector space representation, where traditional clustering algorithms can be applied directly to learn the multi-relational data.

Traditional k-means clustering algorithm is a hill climbing algorithm which is sensitive to its initialization [9][11][12]. The predefined number of clusters, k which is usually assumed to be known priori and the initialization of initial centroids of cluster, will heavily affect the quality of final clusters. The quality of clustering and the time complexity in terms of the number of iterations required to converge also depends on the initial selection of centroids. As different initialization in k-means clustering algorithm will produce different results, which affect the performance of Dynamic Aggregations of Relational Attributes, it is essential to combine and generalize these results produced by different initialization to form one single consensus result of higher stability using clustering ensembles.

Thus, the objectives of this paper are, to design and propose the complete framework of the ensemble approach to learning multi-relational data; to implement a Genetic Algorithm based clustering algorithm in order to find the best initialization for each k-means clustering applied; and to assess the performance of the clustering ensembles by using a C4.5 classifier and Naïve Bayes classifier based on different distance metrics and different fitness function (i.e., single and multi-objectives). The rest of this paper is organized as followed. Section 2 explains some related works. Section 3 discusses the Multi-Objectives Genetic Algorithm Hybridization used in this paper. Section 4 discusses the experimental setup and Section 5 discusses the experimental results obtained. Section 6 concludes this paper.

2 Related Works

2.1 Multi-Relational Data Mining

A typical multi-relational database consists of multiple relations as tables and these tables are associated to each other through primary and foreign keys. In a one-to-many relationship set of tables, a record stored in the target table corresponding to multiple records stored in the non-target table. One of the famous approaches in learning multi-relational data is known as inductive logic programming [13]. The term for

inductive logic programming was first introduced by Muggleton [14]. It uses logic for representation. Based on the background knowledge provided, it constructs predicates for logic representations and make hypotheses base on the syntactic predicates [6]. However, most of the inductive logic programming based approaches are usually inefficient for databases with complex schemas, not appropriate for continuous values and not capable of dealing with missing values [15].

Another common approach is propositionalization [7] that captures and stores relational representation in propositional form. These propositional forms are known as new features and are usually stored as attributes in a vector form. Although propositionalization approaches have some inherent limitations, such as learning recurrent definitions, it has advantages over traditional inductive logic programming based approaches by allowed the use of propositional algorithms including probabilistic algorithms while learning relational representations [7].

Alfred recently introduced Dynamic Aggregation for Relational Attributes approach to summarize the entire contents of non-target tables before the target table can be processed for knowledge discovery [8]. It contains three stages i.e. data preprocessing, data transformation and data summarization. Data are first preprocessed by discretizing continuous values into categorical values. Then, a theory from information retrieval is borrowed, to transform the multi-relational databases. Records from non-target table are transformed into "bag of words", so that frequency-inverse document frequency matrix can be produced.

2.2 Clustering

After where data is transformed from multi-relational data representation to vector space representation, a traditional clustering algorithm can then be applied. Data mining can be generalized to supervised learning and unsupervised learning. Clustering is an unsupervised learning algorithm where class labels are not available during training phase [16]. Clustering can be categorized to two main categories: hierarchy and partition [10][11][17][18]. Hierarchy clustering will produce nested clusters tree. In contrast with hierarchical clustering, a record can be grouped from one cluster to another cluster in different iterative of partition clustering. *k*-means clustering is a partition clustering algorithm [19]. The number of clusters, k is predefined. For each N record instances, we will calculate the vector distance between the records to each of the centroid of clusters using Euclidean distance, $dist$, as follows,

$$dist = \sqrt{\sum_{i=1}^{d}(x_i - y_i)^2} \tag{1}$$

where d is the maximum feature dimension, x is the record instance and y is the centroid of interest. The record instance will be assigned to the nearest centroid, which is the centroid with lowest Euclidean distance.

Alternatively, the distance between the records to each of the centroid of clusters can also be calculated using the cosine similarity, as follows,

$$\cos(x, y) = \frac{x \cdot y}{\|x\|\|y\|} \quad (2)$$

where $\cdot$ indicates the vector dot product, $x \cdot y = \sum_{k=1}^{n} x_k y_k$, and $\|x\|$ is the length of vector x, $\|x\| = \sqrt{\sum_{k=1}^{n} x_k^2} = \sqrt{x \cdot x}$. Through finite number of iterations, *k*-means algorithm is guaranteed to converge to a local optimum. *k*-means algorithm is sensitive to the centroids initialization. It is common to run *k*-means clustering with different number of clusters and different centroids initialization. After clustering algorithm is applied, clustering validity can be used to evaluate the quality of the final clustering result. The clustering validity index has been defined as combination of compactness and separation [20]. Compactness is the measurement of distance between records of the same cluster. Lower distance between records of the same cluster should have better validity. Separation is the measurement of distance between records of different cluster. Records of different cluster should separate as far as possible with each other. The Sum of Squared Error (SSE) is the summation of squared distance for each record in the data set with other records [21]. It is defined as,

$$SSE(X, Y) = |X - Y| \quad (3)$$

where X and Y are vector of two records in the vector space. Davies-Bouldin Index (DBI) is an internal validity index [20]. It is defined as,

$$DBI = \frac{1}{c} \sum_{i=1}^{c} \max_{i \neq j} \left\{ \frac{d(X_i) + d(X_j)}{d(c_i, c_j)} \right\} \quad (4)$$

where c is the total number of clusters, i and j are the cluster labels in comparison, $d(X_i)$ and $d(X_j)$ are records in cluster i and j to their respective centroids, $d(c_i, c_j)$ is distance between centroid i and j. Lower value of DBI indicates better clustering result. Entropy (Ent) is an external validity index [20]. It is the measurement of purity of clusters' class labels. Lower value of Entropy indicates lower impurities within the same cluster. First, we calculate the class distribution of objects in each cluster as follows,

$$E_j = \sum_i p_{ij} \log(p_{ij}) \quad (5)$$

where j is the cluster, i is the class. p_{ij} is the probability of occurrences of class i in cluster j. Total Entropy is calculated as follows,

$$E = \sum_{j=1}^{m} \frac{n_j}{n} E_j \quad (6)$$

where n_j is the size of cluster j, m is the number of clusters, and n is the total number of records in the data set.

2.3 Multi-Objectives Optimization

Real life optimizations are mostly multi-objectives and these objectives are mostly contradictory. The goal of multi-objectives optimizations is to search for solutions

that optimize a number of functions to satisfy more than one constraint (Wahid, Gao, & Peter, 2014). Traditional genetic algorithms [22][23][24] can be accommodated with specialized multi-objectives function in order to provider higher solutions diversity. There are generally two approaches to multi-objectives optimization. The first general approach is either combines individual functions into one composite function or moves all objectives to one constraint set. Combining individual functions is also known as weighted sum approaches (Wen, Li, Gao, Wan, & Wang, 2013), where a weight w_i is assigned for each normalized objective function, $f_i(x)$ so that the multi-objectives problem is converted to a single objective problem:

$$f(x) = w_1 f_1(x) + w_2 2(x) + \cdots + w_i f_i(x) \tag{7}$$

In clustering for instance, there are many functions that can be used to validate the cluster quality such as SSE, DBI and entropy as described in Section 2.2 above. These individual cluster quality validity functions can be normalized and then combined using equal weighted sum to convert it to a single minimization function.

3 Multi-Objectives Genetic Algorithm Hybridization

Since *k*-means algorithm is sensitive to initialization and would fall into a local optima, rather than running it for multiple times using random initialization, it has been hybridized with a genetic algorithm [22][23][24] in order to find the global optima. In this paper, a binary encoding is used with chromosome length = N where N is the total number of records. Each gene within the chromosome represents an instance of records of the data set. The value of each gene represents whether that particular instance of records is an initial centroid of clusters. For each number of clusters, k the population size of 100 is used and is let to evolve for 100 generations. Each chromosome represents a possible solution, which is a set of initial centroid of clusters. After k-means algorithm run is completed for each chromosome in the first generation, the chromosome quality will be evaluated using the individual fitness function previously and multi-objectives function as described below.

Individual fitness functions described in Section 2.2, can also be normalized, combined according to weighted sum approach into one optimization objective. The normalization of each of the single objective function is done based on per generation basis in the genetic algorithm,

$$f_{normalized}(x) = \frac{f_i(x) - f_{\min}(x)}{f_{max}(x) - f_{min}(x)} \tag{8}$$

where $f_i(x)$ is the fitness value of the individual, $f_{max}(x)$ and $f_{min}(x)$ is the maximum and minimum fitness value in the generation respectively. The multi-objectives fitness function can then be defined as,

$$f^*(x) = w_1 f_1(x) + w_2 f_2(x) + w_3 f_3(x) \tag{9}$$

where $f^*(x)$ is the computed weighted sum multi-objectives fitness value of the individual, $f_1(x)$, $f_2(x)$ and $f_3(x)$ are normalized value for sum of squared error, Davies-Bouldin index and cluster entropy respectively, and w_1, w_2 and w_3 are the weight applied to respective normalized value, where $w_1 = w_2 = w_3 = 1$.

4 Experimental Setup

The main objective of the experiment that is conducted involves the task of finding the best initialization with respect to the number of clusters, proximity distance measurements, fitness functions, and classifiers used for the evaluation. There are two main parts in this experiment. In the first part, the data will be clustered based on given number of clusters, *k*. The main task here is to determine the best number of clusters in order to summarize the transformed data. Let the number of rows in a data set be n then the ranges of *k* will be from 2 to $2\sqrt{n}$, with increment of 1. For example, there are 189 rows of data within Mutagenesis dataset, 2 times of square root of 189 is 27. We further extended 27 to 30 to allow better view of the output results in term of accuracy. The clustering results are then appended into the target table as new additional feature. Then the table is fed into the classifiers (e.g., C4.5 and Naïve Bayes classifiers with 10-fold CV) in order to evaluate the predictive performances.

In the second part of the experiment, the results from the first part of the experiment with number of clusters, *k* from 2 to maximum test range is collected. The cluster ID which is determined from the first part of this experiment will become the features of the second part of the experiment. The second part of the experiment consists purely of categorical data as cluster ID is categorical value. Same range of *k* is used. After clustering, the results are fed into WEKA using the same classifier. Results of the second part are then compared with the first part of the experiment.

For each number of clusters, *k*, a population size of 100 is used and there is 100 generations. Each chromosome represents a set of initial centroid of clusters. During reproduction, the chromosomes will undergo uniform crossover and mutation. For each gene of the chromosome, they will be tested with a crossover rate whether to exchange the information. After the offspring is produced, each of the genes of the offspring chromosome will be tested with mutation rate whether to flip the bit. To maintain the number of true bits which represent the number of clusters, when a bit is flipped, another bit of opposite value is also randomly chosen and flipped.

The experiment is first executed with genetic algorithm using sum of squared error (SSE) as measurement of fitness function which will be used for selection. The experiment is then repeated using Davis-Bouldin Index (DBI) and Cluster Entropy (Ent) as fitness function. After that, the experiment is repeated again using a normalized multi-objectives (MO) fitness function, which is the combination of all previous three fitness function earlier.

Three Mutagenesis and Three Hepatitis datasets were used, namely B1, B2, B3 and H1, H2, H3 respectively. The number of clusters, *k* used in Mutagensis datasets starts from *k*=2 to *k*=30. The number of clusters, *k* used in Hepatitis datasets starts from *k*=2 to *k*=50. Each of the dataset was executed in the experiment using 3 differ-

ent single fitness function and 1 combined and normalized multi-objectives fitness function.

5 Results and Discussion

Six different datasets i.e. B1, B2 and B3 from mutagenesis and H1, H2 and H3 from hepatitis have been used. Each dataset is then summarized using different combination of distance measurement (e.g., Euclidean distance and Cosine) and genetic algorithm fitness function (e.g., SSE, DBI, Ent, MO) to produce sets of 48 results. Each result set are then fed into two different classifier, C4.5 and Weka Naïve-Bayes classifiers, which consequently produce the 96 average predictive accuracy increment tabulated in Table 1.

Table 1. Increment of average predictive accuracy in percentage generated by different from classifiers using summarized datasets with clustering ensembles of different combination of experimental settings.

Weka Classifier	Dataset \ Fitness Function, Distance Measurement	Sum of squared error		Davies-Bouldin index		Cluster entropy		Multi-objectives	
		Euclidean	Cosine	Euclidean	Cosine	Euclidean	Cosine	Euclidean	Cosine
C4.5	B1	1.85	1.62	-0.46	0.97	3.59	0.28	2.09	2.60
	B2	0.83	-0.07	-0.02	1.78	2.44	2.84	0.00	1.41
	B3	-0.38	-0.44	4.51	0.30	3.93	3.19	4.12	0.33
	H1	2.15	2.76	-9.4	6.43	2.44	2.03	2.20	2.97
	H2	-0.02	-0.01	-0.09	0.00	0.41	1.25	0.00	0.66
	H3	-0.08	0.00	-0.09	0.00	1.46	0.07	0.07	0.15
Naïve Bayes	B1	0.11	10.11	1.08	-0.49	-0.49	0.97	0.11	1.24
	B2	0.38	-0.97	0.49	-0.81	2.37	1.35	1.73	-0.38
	B3	0.32	0.27	0.97	0.16	3.18	2.21	0.91	0.60
	H1	3.59	1.96	2.45	4.33	2.35	3.59	2.75	5.77
	H2	3.62	1.10	3.59	0.71	8.31	3.76	6.91	3.54
	H3	6.18	-0.30	5.13	0.39	9.10	-1.00	9.55	0.70

In term of classifier used to generate predictive accuracy, Weka C4.5 classifier is able to achieve increment in average predictive accuracy as high as 6.43% with 32 results set of positive increment and 16 results set of no improvement or decreased average predictive accuracy, which is able to conclude that 66.66% of the summarized data can perform better on a Weka C4.5 classifier. On the other hand, Weka Naïve Bayes classifier is able to achieve increment in average predictive accuracy as high as 10.11% with 41 results set of positive increment and 7 results set of no improvement or decreased average predictive accuracy, which brings to a conclusion that 85.41% of the summarized data can perform better on a Weka Naïve Bayes classifier. From this point of view, Weka Naïve Bayes classifier will benefits from a summarized data more than a Weka C4.5 classifier does.

Table 2. Average predictive accuracy increment in percentage based on different classifier used and fitness function used in the genetic algorithm.

Fitness Function \ Classifier	Weka C4.5	Weka Naïve Bayes	Overall
Sum of squared error	0.67	2.19	1.43
Davies-Bouldin index	0.32	1.5	0.91
Cluster entropy	1.99	2.97	2.48
Multi-objectives	1.38	2.78	2.08

Table 3. Average predictive accuracy increment in percentage based on different classifier used and different summarized dataset used.

Dataset \ Classifier	Weka C4.5	Weka Naïve Bayes
Mutagenesis (B1, B2, B3)	1.55	1.05
Hepatitis (H1, H2, H3)	0.63	3.67

The table above concluded the results from point of view of performance of different fitness function used in the genetic algorithm to optimize the initialization of initial centroids of the k-means algorithm. From the results, it can be concluded that cluster entropy has the highest performance in optimizing the initial centroids of the k-means clustering algorithm, with overall average increment of predictive accuracy of 2.48%, followed by multi-objectives with 2.08%. This is most probable caused by that class label has been used as part of evaluation in the cluster entropy fitness function, which favors both classifiers.

The table above concluded the results using another perspective, to compare which summarized datasets have better performance in term of average predictive accuracy increment according to different classifier used. It can be concluded that summarized mutagenesis datasets have better performance for Weka C4.5 classifier but summarized hepatitis datasets have better performance for Weka Naïve Bayes classifier.

6 Conclusion

In this paper, a framework of ensemble approach to learning relational data is proposed and designed. A genetic algorithm based *k*-means clustering algorithm has been implemented in clustering relational data to find the optimal set of solutions for each number of clusters, k. It can be concluded that the combination of Euclidean distance has the better performance over cosine similarity for mutagenesis datasets and Weka C4.5 classifier, but cosine similarity has better performance over Euclidean distance for hepatitis datasets and Weka C4.5 classifier. On the other hand, it was found that for Weka Naïve Bayes classifier, where cosine similarity has better performance for mutagenesis dataset and Euclidean distance has better performance for hepatitis dataset. The weight for each individual objective function in the multi-objectives optimization process in this work is not properly tuned as it is not an easy task even for expert with domain knowledge. To address this problem in the future, the weights of each individual function can also be optimized using evolutionary algorithms. The purpose of doing so is to find the optimum combination of weights for each individual

function along the evolutionary algorithms of multi-objectives optimization in order to find the best set of initial centroids. Pareto approaches can also be implemented to compare the performance of cluster ensemble with a weighted sum approach.

In future works, the processes in the transformation stage may be integrated with the summarization stage, such as feature aggregation or feature selection using an evolutionary approach, together with the optimization of initial centroids of *k*-means clustering; so that the best set of features contributing to the highest information gain can be selected, thus indirectly ignoring the insignificant tuples in the non-target table.

References

1. R. Cattral, F. Oppacher, and K.J.L. Graham: Techniques for evolutionary rule discovery in data mining, Conference on Evolutionary Computation, 2009. pp. 1737-1744.
2. Lei Xu, Chunxiao Jiang, Jian Wang, Jian Yuan, and Yong Ren: Information Security in Big Data: Privacy and Data Mining, IEEE, 2014, pp. 1149-1176.
3. Saso Dzeroski: Relational Data Mining, Data Mining and Knowledge Discovery Handbook, Oded Maimon and Lior Rokach, Eds.: Springer US, 2010, pp. 887-911.
4. Ping Ling and Xiangsheng Rong: Double-Phase Locality Sensitive Hashing of neighborhood development for multi-relational data, 13th UK Workshop on Computational Intelligence (UKCI), pp. 206-213, 2013.
5. Urvashi Mistry and Amit R Thakkar: Link-based classification for Multi-Relational database, Recent Advances and Innovations in Engineering (ICRAIE), pp. 1-6, 2014.
6. Wei Zhang: Multi-Relational Data Mining Based on Higher-Order Inductive Logic, WRI Global Congress in Intelligent Systems, Xiamen, pp. 453-458, 2009.
7. Dan Roth and Wen-tau Yih: Propositionalization of Relational Learning: An Information Extraction Case Study, 17th International Joint Conference on Artificial Intelligence, Seattle, 2001.
8. Chung Seng Kheau, Rayner Alfred, and Lau Hui Keng: Dimensionality Reduction in Data Summarization Approach to Learning Relational Data, Intelligent Information and Database Systems, LNCS, Berlin, Germany: Springer, 2013, pp. 166-175.
9. Bin Lu and Fangyuan Ju: An optimized genetic K-means clustering algorithm, International Conference on Computer Science and Information Processing, 2012, pp. 1296-1299.
10. Taoying Li and Yan Chen: A Weight Entropy k-means Algorithm for Clustering Dataset with Mixed Numeric and Categorical Data, Fifth International Conference on Fuzzy Systems and Knowledge Discovery, Shandong, pp. 36-41, 2008.
11. Xiaohui Cui, Thomas E. Potok, and Paul Palathingal: Document Clustering using Particle Swarm Optimization, Swarm Intelligence Symposium 2005, IEEE, pp. 185-191, 2005
12. R.F. Abdel-Kader: Genetically Improved PSO Algorithm for Efficient Data Clustering, 2nd International Conference on Machine Learning and Computing (ICMLC), pp. 71-75, 2010
13. Nileshkumar D. Bharwad and Mukesh M. Goswami: Proposed efficient approach for classification for multi-relational data mining using Bayesian Belief Network, 2014 Int Conf on Green Computing Communication and Electrical Engineering, pp. 1-4, 2004.
14. Stephen Muggleton: Inductive Logic Programming, New Generation Computing, 8(4): 295-318, 1991.

15. Jingfeng Guo, Lizhen Zheng, and Tieying Li: An Efficient Graph-based Multi-relational Data Mining Algorithm, International Conference on Computational Intelligence and Security, pp. 176-180, 2007.
16. Dipankar Dutta, Paramartha Dutta, and Jaya Sil: Data Clustering with Mixed Features by Multi Objective Generic Algorithm, 12th International Conference on Hybrid Intelligent Systems, Pune, pp. 336-341, 2012.
17. Neepa Shah and Sunita Mahajan: Document Clustering: A Detailed Review, International Journal of Applied Information Systems, vol. 4, pp. 30-38, October 2012.
18. Chun-Ling Chen, Frank S.C. Tseng, and Tyne Liang: An integration of WordNet and fuzzy association rule mining for multi-label document clustering, Data & Knowledge Engineering, 69(11):1208-1226, 2010.
19. David Pettinger and Giuseppe Di Fatta: Space Partitioning for Scalable K-Means, 9th International Conference in Machine Learning and Apps (ICMLA), pp. 319-324, 2010.
20. Erendira Rendon, Alejandra Arizmendi Itzel Abundez, and Elvia M. Quiroz: Internal versus External cluster validation indexes, International Journal of Computers and Communications, 5(1):27-32, 2011.
21. M. Bilal, S. Masud, and S. Athar: FPGA Design for Statistics-Inspired Approximate Sum-of-Squared-Error Computation in Multimedia Applications,Circuits and Systems II: Express Briefs, IEEE Transactions, 59(8):506-510, 2012.
22. Melanie Mitchell: An Introduction to Genetic Algorithms. London, England: MIT Press, 1999.
23. Noraini Mohd Razali and John Geraghty: Genetic Algorithm Performance with Different Selection Strategies in Solving TSP, Proceedings of the World Congress on Engineering 2011, vol. II, London, 2011.
24. Abdul Wahid, Xiaoying Gao, and Andreae Peter: Multi-View Clustering of Web Documents using Multi-Objective Genetic Algorithm, 2014 IEEE Congress Evolutionary Computation (CEC), pp. 2625 – 2632, 2014.
25. Xiaoyu Wen, Xinyu Li, Liang Gao, Liang Wan, and Wenwen Wang: Multi-objective genetic algorithm for integrated process planning and scheduling with fuzzy processing time, 2013 Sixth International Conference on Advanced Computational Intelligence (ICACI), pp. 293-298, 2013.
26. Abdullah Konak, David W. Coit, and Alice E. Smith: Multi-objective optimization using genetic algorithms: A tutorial, Reliability Eng and System Safety, 9(9):992-1007, 2006.
27. Fatimah Sham Ismail, Rubiyah Yusof, and S Muhd Masduqi Waqiyuddin: Multi-objective optimization problems: Method and application, 2011 4th International Conference on Modeling, Simulation and Applied Optimization (ICMSAO), pp. 1-6, 2011.
28. Nafissa Zeghichi, Mekki Assas, and Leila Hayet Mouss: Genetic Algorithm with Pareto Fronts for Multi-Criteria Optimization Case Study Milling Parameters Optimization, Software, Knowledge Information, Industrial Management and Applications (SKIMA), 2011 5th International Conference, Benevento, pp. 1-5, 2011.
29. K. Atashkari, N. NarimanZadeh, A. R. Ghavimi, M. J. Mahmoodabadi, and F. Aghaienezhad: Multi-objective optimization of power and heating system based on artificial bee colony, Innovations in Intelligent Systems and Applications (INISTA), 2011 International Symposium, Istanbul, 2011, pp. 64-68.