International Journal of Database Theory and Application
Vol.10, No.1 (2017), pp.57-66
http://dx.doi.org/10.14257/ijdta.2017.10.1.06

Temporal and Spatial Association Rules Strong Mining Algorithm Based on Hierarchical Reasoning Parameters

Zhang Xuewu[1]

[1]*College of Computer Science and Engineering, Changshu Institute of Technology, Changshu, Jiangsu 215500, China*
E-mail: zhangxuewu4116@163.com

Abstract

Such problems as premature convergence and local optimal solution universally exist in the application of traditional genetic algorithm to the association rules mining, so a lot of time is needed for extracting the useful strong association rules. In order to conquer these disadvantages, the adaptive variation rate is introduced in this paper and the method for the operator selection during the genetic process is improved in order to specifically improve the traditional genetic algorithm, and the improved association rules mining method is used to analyze the power transformation equipment defect data. The example comparison shows that the improved genetic algorithm can significantly reduce the rule discovery calculation complexity and improve the association rules mining efficiency.

Keywords*: Data mining; genetic algorithm; Association rules; Adaptive Variation rate; Premature convergence*

1. Introduction

Data mining is one of the main research directions in the field of the international database and information decision research, and is mainly used to research and develop relevant methods, theories and tools for discovering significant and practical knowledge in massive data. As one of the important research contents of data mining, the association rules mining aims at presenting the hidden association in the transactional databases. Along with data accumulation, how to mine and discover the association among many data sets from different data sources has been concerned by more and more scientific workers [1-3]. Apriori algorithm is a typical association rules mining algorithm, and many association rules mining methods and the variation thereof are based on Apriori algorithm thoughts. Apriori algorithm needs to scan the database for several times, so it is not applicable to the rules mining task in large-scale databases due to lots of input and output requests and excessive calculation complexity [4-5].

GA (Genetic Agorithm) is an intelligent algorithm based on the biological evolution theory and the global random search thought of the molecular genetics. With high randomness, strong robustness and parallel calculation capability, this algorithm can rapidly and effectively find the global optimal solution, thus becoming an efficient approach for discovering rules in large-scale data set. At present, certain progress has been obtained in the research on the association rules mining method based on the genetic algorithm, and some achievements have been obtained for the classifiers based on the genetic algorithm. However, few researches have been implemented for combining the genetic algorithm and the association rules mining for practical data analysis [6-8]. In this paper, the genetic algorithm is adopted to discover the association rules, and the disadvantages of the genetic algorithm is also specifically improved in order to apply the genetic algorithm to the association rules mining of the electrical equipment.

ISSN: 2005-4270 IJDTA

2. Association Rules Mining Algorithm

2.1. Association Rules

The concept of the association rules is firstly proposed by R. Agrawal for describing the purchasing pattern of the supermarket customers in order to discover the association of various attributes in the transactional databases [9-10]. The association rules of a transactional database can be described as follows:

$I = \{i_1, i_2, \ldots, i_n\}$ is assumed as the item set, and the task related data *D* is the database service set, and each transaction *T* is the item set, namely $T \subseteq I$. Each transaction has one identifier called as *Tid*. A is assumed as an item set, transaction *T* includes *A*, namely $A \subseteq T$. The association rule is an implication similarly to $A \Rightarrow B$, wherein $A \subset I$, $B \subset I$ and $A \cap B = \Phi$. Rule $A \Rightarrow B$ is true in transaction set *D*, and has a support degree as *s*, wherein *s* is the percentage for transactions in D to contain $A \cup B$ (union set of *A* and *B*, or *A* and *B*), and the probability thereof is $P(A \cup B)$. Rule $A \Rightarrow B$ has a confidence coefficient *c* in transaction set *D*, wherein *c* is the percentage for *D* to contain both *A* and *B* transactions, and this is conditional probability $P(B|A)$. Support degree *c* and confidence coefficient *s* can be calculated according to Formulae (1) and (2).

$$\mathrm{Sup}(A \Rightarrow B) = P(A \cup B) \tag{1}$$

$$\mathrm{Con}(A \Rightarrow B) = P(B \subseteq T | A \subseteq T) \tag{2}$$

Meanwhile, the rule able to meet the minimum support degree threshold value (minsup) and the minimum confidence coefficient threshold value (minconf) is called as the strong association rule. If the support degree of item set *A* is not less than the minimum support degree threshold value, then *A* is called as frequent item set; or else, *A* is called as non-frequent item set [10-12].

2.2. Genetic Algorithm

GA (Genetic Algorithm) is an adaptive algorithm for obtaining the global optimal solution through simulating the biological evolution and heredity under natural environment. This algorithm, proposed by Professor Holland J of the University of Michigan for the first time in 1975, has unique advantages in the aspect of solving the global optimization problems with large search space, multiple peak values, nonlinearity and high complexity [13-15]. The algorithm mainly aims at representing the parameters of the problem to be solved as genes through binary coding or decimal coding (or other system coding), wherein multiple genes form a chromosome, also called as an individual, and many individuals experience the evolution process through selection, intersection and variation similarly in natural environment for continuous iteration updating till the optimal solution is found [16-18]. In order to solve practical problems, it is necessary to pay attention to six key factors during the application of the genetic algorithm, such as coding, fitness function, selection algorithm, intersection algorithm, variation algorithm and control parameters.

3. Association Rules Algorithm Based on Improved GA

The traditional genetic algorithm has the disadvantage of premature convergence, so all individuals may stop evolution at a local optimal solution or the search process is endless due to the elimination of the global optimal solution during the evolution process. In order to solve the two problems, the adaptive variation rate is introduced in this paper to avoid excessive variation or run into local optimal solution. Additionally, the selection method based on individual fitness ranking is adopted in the operation selection calculation of the

genetic algorithm in order to avoid the rapid convergence of the individuals with high fitness and maximally control the diversity and the balance property of all individuals during the convergence process.

3.1. Adaptive Variation Rate

Variation rate *Pm* in the traditional genetic algorithm is a constant value. When *Pm* is a small value in the initial evolution period, the variation link only has a small influence on the whole chromosome complex and is favorable for generating a new gene; when *Pm* is a relatively large value in later evolution period, excessive variation will gradually damage good genes in the chromosome complex and the algorithm may not be converged to the optimal solution. Therefore, it is necessary to adopt an adaptive variation rate method in the evolution process in order to constantly correct the variation rate, as shown in Formula (3).

$$p_m^{(n+1)} = \lambda p_m^0 \sqrt{\frac{\sum_{i=1}^{m} (f_{\max}^{(n+1)} - f_i^{(m)})^2}{\sum_{i=1}^{m} (f_{\max}^{(n)} - f_i^{(n)})^2}} \tag{3}$$

Where P_m^n and $P_m^{(n+1)}$ are respectively the variation rates of the *nth* iteration and the *(n+1t)th* iteration; $f_i^{(m)}$ is the fitness of individual i at the *mth* iteration; $f_{\max}^{(n+1)}$ is the maximum fitness of all individuals at the *(n+1t)th* iteration; $f_i^{(n)}$ is the fitness of individual i at the *nth* iteration; m is the individual quantity; λ is the adjustment coefficient. The genes generated in this way will effectively reduce the convergence time, thus favorable for generating new genes to prevent the whole iterative evolution process from running into the local optimal solution and strengthening the performance of the traditional genetic algorithm.

3.2. Improved Selection Operator

In order to reduce algorithm convergence time and improve search efficiency as well as prevent the whole search process from running into the local optimal solution and having search blind, it is necessary to divide the fitness of the gene cluster during each iteration into four gene subsets with the same size according to advantages and disadvantages, and duplicate two copies of the optimal gene subsets and eliminate the poorest gene subset before next iteration. As mentioned above, the specific steps for the gradual elimination during the selection process are as follows:

Step 1: Rank according to the individual fitness;

Step 2: Duplicate two copies of the first 1/4 individuals in the sequence, duplicate one copy of the individuals ranked as 1/4~1/2 part, and take them as the input for the next round of selection;

Step 3: Reserve the individuals ranked as 1/2~3/4 parts and take them as the input for the next round of selection;

Step 4: Eliminate the last 1/4 individuals for the next round of selection and evolution.

3.3. Association Rules Mining Algorithm Based on GA

3.3.1. Genetic Algorithm Coding

How to code is the basic problem in the genetic algorithm, and an applicable coding method will facilitate the evolution process. Such association rule as $A1 \cap A2 \cap \ldots \cap An \Rightarrow B1 \cap B2 \cap \ldots \cap Bn$ is described as follows: the probability for the concurrence of $A1 \cap A2 \cap \ldots \cap An$ and $B1 \cap B2 \cap \ldots \cap Bn$ is less than the minimum

support degree; under the precondition of the occurrence of $A1 \cap A2 \cap \ldots \cap An$, the probability for the occurrence of $B1 \cap B2 \cap \ldots \cap Bn$ is more than the minimum confidence coefficient.

An array is adopted in this paper for coding the association rules. The number of the elements in the array is equal to the number of the attributes in the transactional database, namely column number. The values of the elements represent different categories in each attribute, as shown in Table 1.

Table 1. Element Attributes Code Table

Attribute 1	Attribute 2	……	Attribute n
Category 11	Category 12	……	Category 1n
……	……	……	……
Category n1	Category n2	……	Category nn

In the array, a code with the length of *n* represents one record of the database. *A* [1] represents attribute 1, *A* [2] represents attribute 2, …, *A*[*R*] represents the *Rth* attribute in the database. Each attribute is represented in a numeric coding form, in this way, an array *A*[N] can be used to represent the attribute category corresponding to each element. Additionally, 0 means that present attribute is not associated to other attributes. Such array coding is not only simple and easy to implement, but also favorable for calculating each operator during the evolution process of the genetic algorithm. For example, intersection and variation processes can be rapidly finished through array calculation.

3.3.2. Fitness Function

The association rules mining aims at finding the association rule able to meet the maximum support degree and the minimum confidence coefficient in the database. The evaluation for the association rules strength is determined by the support degree and the confidence coefficient, so the fitness is calculated according to Formula (4).

$$f(x) = a * \mathrm{Sup}(x) + b * \mathrm{Con}(x) \tag{4}$$

Where x is the rule; a and b are the adjustment coefficients ($0 \le a, b \le 1$), and respectively represent the proportions of the support degree and the confidence coefficient in the rule evaluation. When an individual cannot be interpreted as a rational rule, it is necessary to set the values of a and b in order to make the fitness of the individual as 0.

3.3.3. Rule Evaluation and Screening

In order to evaluate whether the rule discovered thereby can meet relevant requirements, an evaluation function is adopted to calculate the support degree and the confidence coefficient of each individual, thus to determine whether this rule is expected by the user. In the iteration calculation process of the genetic algorithm, the evaluation function should be used to evaluate the rules for all the individuals with the fitness more than the specific value. If the rule can meet individual conditions, the support degree is assumed to be more than 0.1 and the confidence coefficient is more than 0.8, then this rule will be stored in the rule base. The redundancy rules in the rule base should be eliminated after algorithm completion, and the rules finally reserved in the rule base are the strong association rules we need.

3.3.4. Description for the Improved Association Rules Algorithm

Step 1: Initialize the chromosome complex, and calculate the support degree and the confidence coefficient;

Step 2: Adopt the fitness function to calculate the fitness of each individual in the chromosome complex;

Step 3: Adopt the improve selection operator to select the individuals to be evolved in present chromosome complex;

Step 4: Execute the intersection operator operations in present chromosome complex;

Step 5: Execute the variation operator operations in present chromosome complex;

Step 6: Generate a new chromosome complex;

Step 7: Compare with the present termination iteration times; if the iteration termination condition can be met, stop the evolution and output the rule base; or else, return to Step 2.

4. Algorithm Application and Analysis

According to the research and design mentioned in above section, the association rule mining method based on improved genetic algorithm (AGA-ARM) is applied to the main substation equipment defect data mining and analysis. Firstly, the original data should be collated and pretreated, and the data set is sourced from the ledgers and the defect information of transformers, circuit breakers, current transformers, voltage transformers and transformer bushings, wherein the legers information includes seven dimensions, namely: power supply bureau, coordinate longitude and latitude, pollution area grade, service period, manufacturer type, equipment voltage, defect equipment type; the defect information includes two dimensions, namely: defect type and defect description keyword, totally 12,356 entries of effective data. The configuration of the experimental platform is as follows: operating system Windows8.1, memory 8GB, CPU Intel i7-3667U. The algorithms applied in above comparison are implemented by C++ language.

Firstly, the data dimensions of the association to be mined should be completely input into the temporary datasheet for pre-coding. The pretreated data dimensions and the processing method are as shown in Table 2.

Table 2. The Dimension and Pretreatment Method of the Dataset

Data Dimension	*Pretreatment Method*
Power Supply Bureau	Covert the names of 19 power supply bureaus into numeric types: ST1-ST19;
Coordinate Longitude and Latitude	According to 3*3 Sudoku, divide the longitude and latitude coordinates into nine areas labeled as L1-L9;
Pollution Area Grade	Mark grades A-E as P1-P5;
Equipment Voltage	110kV, 220kV and 500kV are respectively corresponding to numeric types: V1-V3
Manufacturer Type	Domestic M1; Joint venture: M2; Imported: M3
Service Period	1~5 years: Y1; 6~10 years: Y2; 11~15 years: Y3; 16~20 years: Y4; above 20 years: Y5;
Defect Equipment Type	Transformer: T1; bushing: T2; circuit breaker: T3; current transformer: T4; voltage transformer: T5;
Defect Type	Adopt C1-C19 to mark 19 different defects in the substation equipment defect classification standard;
Defect Type Keyword	Extract the defect descriptions for the keywords of equipment parts and defect causes in the defect data set, totally 78entries, and adopt W1-W79 to mark them.

After coding the attributes and the values thereof according to the pretreatment method shown in Table 2, it is necessary to establish and initialize AGA-ARM model, wherein

the default minimum support degree threshold value is set as 0.1, the minimum confidence coefficient threshold value is set as 0.7, the size of the chromosome complex is set as 200, the probability for intersection operation is set as 0.9 and the probability for variation operation is 0.01. This model is applied to the defect data mining and analysis, and totally 35 effective association rules such as $\{A_1, A_2, \ldots, A_n\} \Rightarrow C_i$ with the defect type as the conclusion are mined, and some mining results are as shown in Table 3.

Table 3. Result Table of the Association Rules Mining of the Defect Dataset

No.	Association Rules Result	Support Degree	Confidence Coefficient
1	{Y4,P2,V2} => C3	0.012	0.485
2	{L3,M2,Y2} => C9	0.0236	0.469
……	……	……	……
n	{P2,V1,Y3} => C11	0.418	0.249

As shown in Table 3, association rule 1 can be described as follows: the probability for the substation equipment with the voltage grade as 220kV in grade-b pollution area during the service period of 16~20 years to have C3 defect (mechanical defect) is 48.5%.

In order to explain the efficiency of the improved genetic algorithm for the association rules mining in massive data, traditional Apriori and FP-Growth are adopted in this paper to respectively establish two traditional mining models for comparison. Specifically, the operation efficiencies of the three models under 10 groups of different support degree threshold values are tested in this paper, and the comparison result is as shown Figure 1.

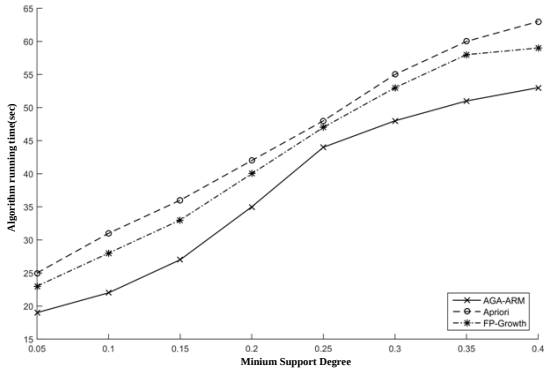

Figure 1. The Comparison of the Application Results of AGA-ARM, Apriori and FP-Growth Algorithms

In the figure, the horizontal axis represents different minimum support degree threshold values and the vertical axis represents the data mining time of traditional Apriori, FP-Growth and AGA-ARM algorithms. According to the comparison of the three curves, Apriori algorithm needs to frequently and repeatedly scan the transactional database during the data set traversing process, thus having the longest operation time; FP-Growth algorithm can effectively compress the memory space structure to avoid repeatedly scanning the transactional database as Apriori algorithm, but many branches are generated during the traversing process to occupy a lot of memory, so FP-Growth algorithm has obvious longer operation time than AGA-ARM algorithm under the same memory condition. Through the genetic parameter setting for improving the algorithm convergence efficiency, AGA-ARM algorithm takes less time to discover the association rules set which is completely consistent with the mining result of the traditional algorithm.

4.1. Real Data Experiment

The density of the established fuzzy data cube is 77%. The calculation is as follows:

$$CubeDensity = \frac{N_{non} - N_{empty}}{N_{cube}} \tag{5}$$

Where, N_{non} is the number of non-empty tuple in the cube; N_{empty} is the number of empty tuple in the cube; N_{cube} is the total of tuple in the cube. The comparing algorithm is set as follows: The discrete association rule mining method referred to in reference [14] is chosen as the comparing algorithm. Reference [15] each rank is provided with the association rule mining method that independently supports threshold and reference [16] association rule algorithm for the minimum mining item set. The parameter is set as follows: the proposed algorithm and reference [16] minimum support threshold have been set to: Rank 1 – 15%, Rank 2 – 5%, Rank 3 – 2%. In reference [15], all minimum support thresholds are set at 5%. In reference [14], the fuzzy set is set at 3. The algorithm experiment data is shown in Figure 2-5 by using the three-dimensional fuzzy data cube.

Figure 2 shows the comparison among the data on variation in the number of frequent item set with the minimum support. As shown in the figure, with an increase of the minimum support, the number of frequent item set of the algorithm tends to decrease, which complies with the fact. A higher threshold will naturally lead to a decrease in the number of item sets that meet the requirements. The number of frequent item set of the proposed algorithm is always bigger than the algorithms compared. The number in reference [14] is more than that in reference [15]. The algorithm in reference [16] has the least number of frequent item set.

Figure 3 shows the experiment data on variation in the number of association rule mining number with the minimum confidence. It shows that the overall variation trend of such experiment data is essentially consistent with that shown in Figure 2. What the difference is that the value of the minimum confidence in references [14] and [15] is 3-5. The association rule mining rule number in reference [15] is more than that in reference [14].

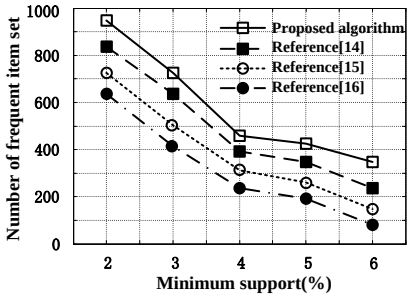

Figure 2. Number of Frequent Item Set – Minimum Support

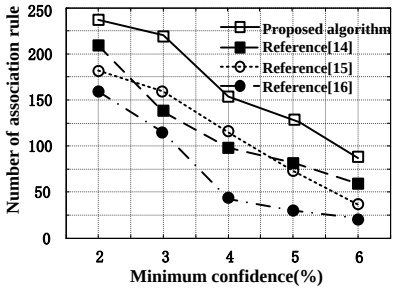

Figure 3. Number of Association Rule – Minimum Confidence

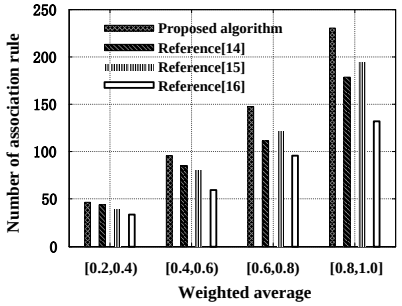

Figure 4. Number of Association Rule – Weighted Average

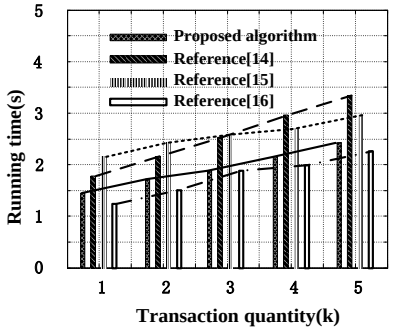

Figure 5. Running Time – Transaction Quantity

Figure 4 shows the variation of association rule with the weighted average. Such rule number tends to increase with an increase of the weighted average. With the same weight, the proposed algorithm has more association rules than the algorithms compared.

Figure 5 shows the variation of algorithm's running time with the transaction quantity. An increase of transaction quantity results in a rise in the calculation time of the algorithm. In general, reference [16] provides higher efficiency in calculation. The proposed algorithm ranks the second place in terms of calculation efficiency but outweighs both algorithms compared in references [14] and [15].

4.2 Synthesized Data Experiment

In this section, the effect of size and density of data cube on algorithm performance is considered. To this end, a three fuzzy data cubes are established by datagen program to set up a synthesis database. Table 4 includes details about such three sparse datasets.

Table 4. Sparse Data Cube

Dimension	Cube density
2（Cube-2） 2 (Cube-2)	12%
3（Cube-3） 3 (Cube-3)	9%
4（Cube-4） 4 (Cube-4)	4%

In the established cube, each dimension has 4 quantitative attributes. Similar to the previous experiment, multi-dimensional data contains 100K experimental records. The experiment design allows for the effect of the minimum support on execution time and frequent item set of the algorithm and the effect of transaction quantity on execution time of the algorithm. The simulation results are included in Table 5.

Table 5 includes the data on synthesized data experiment. As shown in Table 5, an increase in the support leads to a decrease of the calculation time and of the number of the maximal item of the algorithm. In terms of the data cube dimension, an increase in the dimension leads to an increase of the calculation time and of the number of the maximal item of the algorithm under the same minimum support. With the increase of the number of transaction set, both the running time and the calculation time of the algorithm increases. With the same number of transaction set, the calculation time and the number of maximal item of the algorithm increase with an increase in the dimension. Such experiment embodies the extensibility of the proposed algorithm.

Table 5. Synthesized Data Experiment

Dimensions	Index	Minimum support/% (number of transaction set: 60)					Number of transaction set/K (minimum support: 4%)				
		2	3	4	5	6	20	40	60	80	100
Cube-2	M	120	116	112	108	106	93	101	112	125	132
	T	2.3	2.2	2.1	2.0	1.9	1.6	1.9	2.1	2.6	3.3
Cube-3	M	148	145	141	139	137	129	134	141	153	161
	T	3.9	3.7	3.6	3.4	3.2	2.2	2.8	3.6	5.3	5.8
Cube-4	M	159	156	152	150	148	136	143	152	167	174
	T	6.1	6.0	5.8	5.6	5.3	3.9	4.4	5.8	6.7	7.4

5. Conclusion

An improve genetic algorithm is applied in this paper to the association rules mining and analysis of the main substation equipment defect data in order to efficiently extract the significant strong association rules in massive data set. According to the practical algorithm application results, compared with Apriori and FP-Growth algorithms, AGA-ARM algorithm can significantly improve the efficiency of the association rules mining model for massive data set through reducing the database scanning frequency and introducing the evolution search thoughts and the adaptive variation operator.

Acknowledgement

The Foundation of Jiangsu Educational Committee of China under Grant No. 13KJB420001.

References

[1] C. Fu, P. Zhang and J. Jiang, "A Bayesian approach for sleep and wake classification based on dynamic time warping method", Multimedia Tools and Applications, **(2015)**, pp. 1-20.
[2] Z. Lv, "Wearable smartphone: Wearable hybrid framework for hand and foot gesture interaction on smartphone", Computer Vision Workshops (ICCVW), 2013 IEEE International Conference on. IEEE, **(2013)**, pp. 436-443.
[3] Y. Lin, J. Yang and Z. Lv, "A Self-Assessment Stereo Capture Model Applicable to the Internet of Things", Sensors, vol. 15, no. 8, **(2015)**, pp. 20925-20944.
[4] J. Yang, S. He and Y. Lin, "Multimedia cloud transmission and storage system based on internet of things", Multimedia Tools and Applications, **(2015)**, pp. 1-16.
[5] Z. Lv, T. Yin and Y. Han, "WebVR--web virtual reality engine based on P2P network", Journal of Networks, vol. 6, no. 7, **(2011)**, pp. 990-998.
[6] J. Yang, S. He and Y. Lin, "Multimedia cloud transmission and storage system based on internet of things", Multimedia Tools and Applications, **(2015)**.
[7] C. Guo, X. Liu and M. Jin, "The research on optimization of auto supply chain network robust model under macroeconomic fluctuations", Chaos, Solitons & Fractals, **(2015)**.
[8] X. L, Z. Lv and J. Hu, "XEarth: A 3D GIS Platform for managing massive city information", Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA), 2015 IEEE International Conference on. IEEE, **(2015)**, pp. 1-6.
[9] J. Yang, B. Chen and J. Zhou, "A Low-Power and Portable Biomedical Device for Respiratory Monitoring with a Stable Power Source", Sensors, vol. 15, no. 8, **(2015)**, pp. 19618-19632.
[10] G. Bao, L. Mi, Y. Geng and K. Pahlavan, "A computer vision based speed estimation technique for localizing the wireless capsule endoscope inside small intestine", 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), **(2014)**.
[11] X. Song and Y. Geng, "Distributed community detection optimization algorithm for complex networks", Journal of Networks, vol. 9, no. 10, **(2014)**, pp. 2758-2765.
[12] D. Jiang, X. Ying and Y. Han, "Collaborative multi-hop routing in cognitive wireless networks", Wireless Personal Communications, **(2015)**, pp. 1-23.
[13] J. Hu and Z. Gao, "Modules identification in gene positive networks of hepatocellular carcinoma using Pearson agglomerative method and Pearson cohesion coupling modularity", Journal of Applied Mathematics, vol. 2012, **(2012)**.

Authors

Zhang Xuewu, received his Ph.D. degree in geographic information system from Tongji University, China, in 2009. He is currently a lecturer in College of Computer Science and Engineering at Changshu Institute of Technology. His research interest is mainly in the area of data mining, spatiotemporal data analysis and GIS software. He has published several research papers in scholarly journals in the above research areas and has participated in several GIS application software development.

