International Journal of Grid and Distributed Computing
Vol. 9, No. 7 (2016), pp.159-168
http://dx.doi.org/10.14257/ijgdc.2016.9.7.17

Social Network Cloud Structure and Discovery Algorithm Based on DHT

Xiaoshu Zhu*, Junhong Feng, Jie Zhang*

School of Computer Science and Engineering; Guangxi Universities Key Lab of Complex System Optimization and Big Data Processing, Yulin Normal University, Yulin 537000, Guangxi , P.R. China
jgxyzxs@126.com, jgxyfjh@126.com, jgxyzj@126.com

Abstract

Social network has become the important platform for the current social individual to exchange information and access various media. Traditional searching and locating method is faced with the characteristics of the high order correlation and implicit correlation for large-scale users, and user-association multidimensional and heterogeneity. In order to effectively deal with large-scale social network data, and improve the efficiency of user's locating, this article introduces the peer-to-peer distributed searching mechanism with the help of the cloud computing platform. This searching mechanism assigns user a logical identifier, and matches the underlying physical address and the upper users' logical address, so as to build the cloud logical topology structure of social network. This paper designs a K neighbor discovery algorithm. It is used to cluster the nodes according to the features of the user, so as to realize the quickly locating of social network. The performance of the algorithm is analyzed according to the user's searching logic path length of social network, and information amount of routing state. The simulation of the algorithm is evaluated by maintenance costs of average network aggregation coverage and query time. The performance analysis and simulation results demonstrate that social network cloud has good performance and searching efficiency.

Keywords*: Big data storage；Peer-to-peer network；Cluster;DHT*

1. Introduction

Social Networks originates from in the Online Friendship. Users use social network services to organize and maintain the existing social relationships, to discovery new social relations, to present a social person in the virtual network, and to conduct the related social activities[1]. Social network can provide rich images, audio, video and other multimedia data. With the mature of Web2.0 and the widespread popularity of the mobile terminal, it has become one of the most extensive internet applications.[1]

Large online social network "Facebook" has about 1 billion active users[2]. About 60 hours' video data are uploaded in every minute in Website "YouTube" [3]. As can be seen that online social network has a huge number of users and data. These users are not isolated, but connected by social relations and various activities. Online social network has complex correlation, and is a heterogeneous complex network. In order to realize the value of social media, it is key how to accurately and efficiently react and compose of complex network community, to research the mutual dependent and supporting relationship, to deal effectively with large-scale social network data, and to implement high efficiently searching and locating [4].

Xiaoshu Zhu is the corresponding author.

ISSN: 2005-4262 IJGDC

These studies mainly focus on the study of the correlation between social network and location information, the network topology structure of some user's characteristics, so as to improve the locating efficiency of social networks. They do not use the underlying network hierarchy structure and network protocols to solve the problems such as user's high order correlation, implicit correlation, user-association multidimensional and heterogeneity in the process of the social network locating. In order to effectively deal with large-scale social network data, and improve the efficiency of user's locating, this article introduces the peer-to-peer distributed searching mechanism with the help of the cloud computing platform. This searching mechanism assigns user a logical identifier, and matches the underlying physical address and the upper users' logical address, so as to build the cloud logical topology structure of social network. This paper designs a *K* neighbor discovery algorithm. It is used to cluster the nodes according to the features of the user, so as to realize the quickly locating of social network.

2. Related Work

In social network, traditional searching and locating method is faced with the characteristics of the high order correlation and implicit correlation for large-scale users, and faced with user-association multidimensional and heterogeneity. The searching and locating research in the current social network mainly includes as follows. (1)Literature [5-7] designs the information recommendation system. It mines the characteristics of data according to the social network and user's location information, so as to recommend useful information to users. (2) Some queries and services are provided for users to use [8-12]. Literature [13] has researched the connection between the social network and the location information. It has confirmed the correlation between the marks on a geographical location given by the user's friends and those given by user. It has proposed the location recommend system based on the social network. (3)Literature[14] has built the network topology model based on friend group，and put forward the time delay tolerance routing protocol based on cluster structure. The protocol can effectively control the network resources consumption brought by the infection of data copy, on the condition of ensuring high routing performance. (4) The formation of the social network is essentially the user interaction behavior, including creating, maintaining and updating of social relations; and generation and transmission of the content, *etc*. The action of user to user, and the operation of user to the content, has a strong dynamic feature. In the dynamic environment, the application of social media eventually will distribute the contents to the users through the network. The development of online social network has great influence on how for the user to use multimedia. The deployment of data gradually evolve from the traditional distribution pattern by the network center to the edge of network (content providers - user), into the pattern that the media contents transmit on the edge of the network, and conduct archival storage in the center of the network (user - the user). In this pattern, traditional content deployments are faced with great challenges.

3. Topological Structure of Social Network Cloud Coverage and System Architecture Based on DHT

3.1 DHT

DHT(Distributed Hash Table) is the resources locating algorithm adopted by a peer-to-peer network (P2P). It is based on graph theory. The figure plays a more and more important role on effectively describing the data structure of the big data [15]. Hash table, as a kind of data structure, can establish certain corresponding relationship between the storage locations of the files and the keywords of its content. Each keyword corresponds to a unique storage location. In recent years, DHT is introduced into the massive P2P

system, to improve the efficiency of resource locating query and system scalability.

The main thought of DHT algorithm is as follows. (1) Each file index is represented as (*K*, *V*); the key word *K* is the hash value of file name (or other description information of the file); *V* is IP address to store file node (or other description information of the node). *K* and *V* are evenly distributed in the namespace. The index entries of all files (that is, all pair of the (*K*, *V*)) compose a large hash table of file index. Only by inputting *K* value representing the target file, We can find out the addresses of all the nodes to store the file. (2) The above large hash table of file index is divided into some small local block. These small pieces of local hash table are distributed into the participating nodes in the system according to the specific rules. Each node is responsible for the maintenance of a hash table.

3.2 Basic Conception

Definition 1(Social Cluster) Nodes determine the logical position of these data in the social network according to the characteristics of the data to store. These nodes self-organize into social cluster. The maximal value of social cluster is set equal to *M*.

Definition 2（Social Cluster Identifier）Nodes calculate the eigenvector of the local storage data, and gather data object with similar characteristics into a cluster. The maximal cluster is taken as social cluster identifier, in order to determine the position of the node in the social cluster. If the node stores the data of different types of features, the feature subspace stored most of the data objects is saved. The cluster identifier is computed before or in the middle of joining the network or to join the network.

Definition 3(Degree) The number of the adjacent nodes connected directly to the node, its default value is less than or equal to *M*.

Definition 4(Intra-cluster link) The link between the cluster head node and any ordinary node in the cluster.

Definition 5(Inter-cluster link) The link between the cluster head node and any node in the non-adjacent cluster. The distribution C/d^k is utilized to randomly select one point in social cluster space to build inter-cluster link. Where, k is the dimension of the social cluster space; d is the difference of the eigenvector between the two nodes; C is the standard constants whose sum of probability is 1.

Definition 6 (Cluster Coefficient) The probability of which a pair of nodes connected a third node at the same time is connected. It is the ratio of the number of the real existing connected edges, to the total number of edges when all the nodes are fully connected. Given a node $d \in D$， A_d is the local cluster coefficient of d, then $A_d = \frac{\left|\sum_{d \in D} \lambda_d\right|}{l_d(l_d - 1)}$. $A(G)$ is the cluster coefficient of the graph G , then $A(G) = \frac{1}{N}\sum_{d \in D} A_d$. It describes the close degree of graph G.

Note: In A_d， l_d is the total link number of the node $d \in D$. Whereas the neighbor of the node d is a group of nodes $\lambda_d = \{i : P(i, j) = 1\}$.

3.3 Description of System Level Architectural

Chord、CAN、Pastry、Tapestry is the current four typical P2P structured overlay network model. The comparison of several aspects according to the topological structure, routing principle, routing performance aspects, demonstrates that Chord has simplicity, reliability and stable performance. This is the characteristics that other structured overlay network does not have. It has a better scalability and higher searching efficiency. This article is on the basis of Chord to build social network cloud. It is located in the upper layer of Chord. Social network cloud is composed of social network user layer and cloud platform, its layers architecture diagram is shown in Figure. 1.

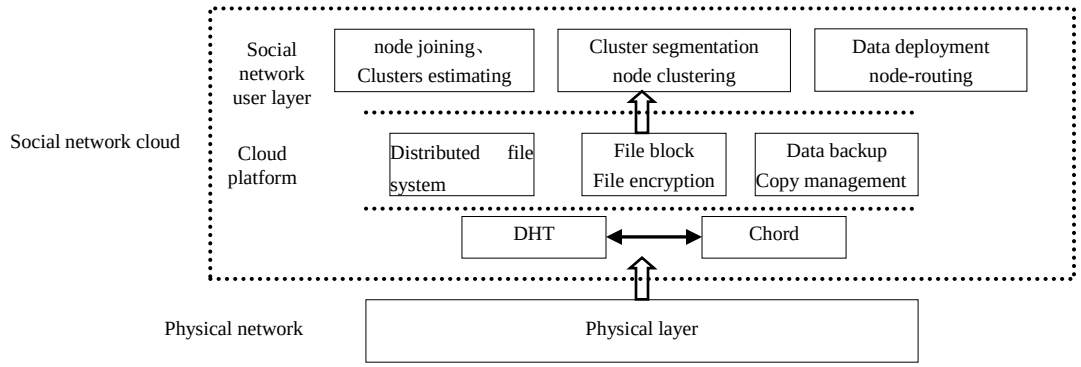


Figure 1. Hierarchical Architecture Diagram of Social

3.4 Description of Overlay Logical Topological Structure

The logical topological structure of social network cloud is a ring structure. As shown in Fig. 2. After joining and leaving some nodes, the whole social network is divided into six clusters (M = 4, k = 3). The ordinary node 11 in Cluster E maintains the intra-cluster link of cluster header node 15. The cluster header node 15 maintains the inter-cluster link between the cluster A and the cluster B.

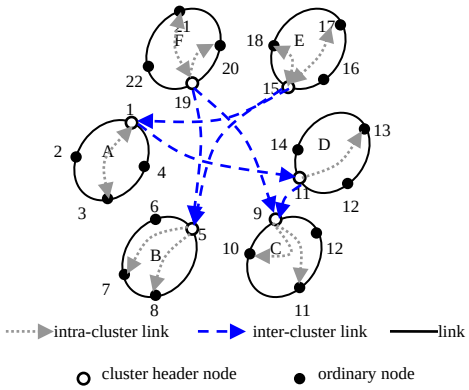


Figure 2. Logical Topological Structure Diagram of Social Network Cloud

4. Social Network Aggregation and Discovery Algorithm Based on DHT

4.1 Network Aggregation and Cluster Segmentation

The eigenvector of data stored by social network user is high-dimensional, whose range is between 50 ~ 300. This may bring about the complexity of network aggregation process. Therefore, dimensionality reduction of eigenvector is needed to conduct. This paper linearizes data clusters in the high-dimensional space into a low dimensional data by means of the cluster segmentation. When the Node Node (a) to join the network, perform the following steps:

Step 1: The node Node(a) send a join request message to the known friend Node Node(f)

in the network. The message contains the eigenvalue of the stored dat.

Step 2: The node Node(f) forward retransmit the request message recursively. At the process of retransmitting, search the cluster *C* contained the eigenvalue of the data to store the Node Node (a) in index area.

Step 3: Judge whether the node Node(a) is to join the cluster *C*, or to perform cluster segmentation. If $mc<M$， the node Node(a) will establish connection with the cluster header node in the cluster *C*, and join the cluster *C*; Otherwise, perform cluster segmentation.

The strategy of cluster segmentation is as follows:

Step 1: The cluster header node Node(m) take a vote in all the intra-cluster nodesin the form of flooding.

Step 2: Select two nodes Node(b) and Node(c) whose the difference of the eigenvalue is maximal.

Step 3: The two nodes Node(b) and Node(c) are respectively taken as the boundary node of new cluster. Assign other nodes in the cluster *C* into the new sub-cluster C_1 and C_2 according to the minimal distance to the two nodes.

Step 4: The cluster header node of each cluster update it's the intra-cluster link and inter-cluster link, and notify all intra-cluster nodes.

The Coding and naming process of social cluster identifier is as follows:

All the bits of the identifier represented the first social cluster in the network are set equal to 0. If the cluster segmentation is triggered, then the cluster is separated into two sub-clusters. The zone bit of the first sub-cluster is set equal to 0; another is 1. The other zone bits are the same as the initial cluster identifier. When more nodes join the network, the sub-clusters obtained by cluster segmentation conduct the same process to acquire the cluster identifier.

4.2 Description of Discovery Algorithm

When the node Node (a) wants to find similar *k* nodes stored data eigenvector, firstly generate a query request message, including query eigenvector *SV*. Perform the following steps:

Step 1: The node Node(a) send a query request to all connected nodes and cluster header node in the cluster contained the node. The eigenvector of query is *SV*.

Step 2: If *k* nodes can be found out in the cluster, then the information of the *k* nodes is sent into the node Node(a). If the number of the found nodes *h* is less then *k*, then perform Step 3.

Step 3: The node Node(a) calculate the cluster identifier according to eigenvector *SV*. Then we trace the other related cluster identifier by means of each bit of the cluster identifier. The cluster header node of the cluster *C* contained the node Node(a) send the query requests recursively. If *k*-*h* nodes can be found out, then the information is sent into the node Node(a). Otherwise, perform Step 4.

Step 4: Calculate *R*, $SV_R=[SV-R_0, SV+R_0]$, R_0 is equal to the average distance between eigenvector *SV* of Node(a) and eigenvector *SV*' of the nodes found in the cluster. The node Node(a) calculate the range of the cluster identifier according to eigenvector SV_R. Then we trace the other related cluster identifier by means of each bit of the cluster identifier. The cluster header node of the cluster C contained the node Node(a) send the query requests recursively, till the query successfully terminate.

5. Performance Analysis and Simulation

This article mainly adopts the logical path length, information amount of routing state two things to analyze the performance of social network cloud. Let *N* represent the

number of nodes; t the number of clusters; s the number of inter-cluster link; M_i the number of the nodes within the i-th cluster. Therefore, $\sum_{i=1}^{t} M_i = N$ is the average number of the nodes, $\overline{M} = N/t$ 。

(1) Logical Path Length $\overline{M}$

The logic path length obtained by performing one resource location or query in social network cloud is analyzed as follows.

Conclusion 1: Social network cloud more optimizes the logic path length. The logic path length in Chord is $\log(N)$ hops; Social network cloud $\frac{(1+\log_2(N/2M))8\ln(3N/M)}{s}$ hops.

Proof: Suppose $L(i,j)$ be the distance between the clusters M_i and M_j, then $L(i,j) = |j - i|$.

The inter-cluster links are generated according to the following probability density function.

$$p(x) = \frac{1}{x\ln(t)}, x \in (1,t)$$

The probability of far links the clusters M_i and M_j is as the following equation.

$$\frac{L(i,j)^{-1}}{\sum_{j\neq i} L(i,j)^{-1}}$$

where,
$$\begin{aligned}\sum_{j\neq i} L(i,j)^{-1} &\leqslant \sum_{y=1}^{t-2} 2y^{-1} = 2\sum_{y=1}^{t-2} y^{-1} \\ &\leqslant 2+2\ln(t-2) \\ &\leqslant 2\ln 3 + 2\ln(t-2) \\ &\leqslant 2\ln(3(t-2)) \\ &\leqslant 2\ln(3t)\end{aligned}$$

When query messages are retransmitted the intermediate nodes whose distance is y inter-cluster hops to the target nodes, we assume CH_y to be inter-cluster hops from the intermediate nodes to the target nodes.

$$\begin{aligned}E\left[CH_y\right] &= \sum_{i=1}^{\infty} \Pr[CHy \geq i] \\ &\leq \sum_{i=1}^{\infty} (1 - \frac{s}{8\ln(3m)})^{i-1} \\ &= 8\ln(3t)/s\end{aligned}$$

Assume H to be the total hops of object searching algorithm, then

$$H = \sum_{y=0}^{\log_2(t/2)} CH_y$$

$$\begin{aligned}E(H) &\leq \frac{(1+\log_2(t/2))8\ln(3t)}{s} \\ &= \frac{(1+\log_2(N/2M))8\ln(3N/M)}{s}\end{aligned}$$

(2) Information Amount of Routing State

Conclusion 2: each node of social network cloud averagely maintain less routing (neighborhood) state information, and have lower expenditure to join and exit the node.

Proof: The routing information of each node in Chord is $n_1 = O(\log N)$; the routing information of each ordinary node in social network cloud is average $O(\log M)$; the routing information of each cluster header node is $O(\log M)$; the average routing information of nodes is

$$n_2 = \frac{(N - \frac{N}{M})O(\log M) + \frac{N}{M}(M + s)}{N} = (1 - \frac{1}{M})O(\log M) + \frac{1}{M}O(M + s) = (1 - \frac{1}{M})O(\log M) + O(1 + \frac{s}{M})$$

As the increasing velocity of the nodes number N is much larger than the intra-cluster nodes number M and the number of inter-cluster link s, $n_1 > n_2$. It is visible that as social network cloud divides nodes into several levels, each node maintains less routing information.

5.2 Simulation

In order to analyze the performance of social network cloud, the article simulates a dynamic social network. Two aspects, maintenance costs of average network aggregation coverage and query time, are utilized to evaluate and simulate the performance of social network cloud. When the number of nodes attains N, nodes will join to the network at six nodes per second according to poisson distribution. The life cycle of nodes is set as TTL=[10,30]min. The number of nodes contained in each cluster is in the interval [5, 10]. Each node stores 0~10 data. After network size reaches a steady state by go through some nodes to join or leave, the nodes number N=256. Therefore, about total 2000 files are

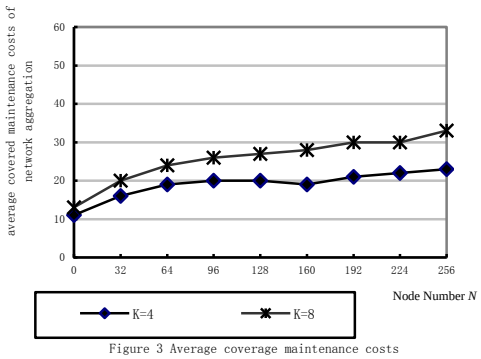
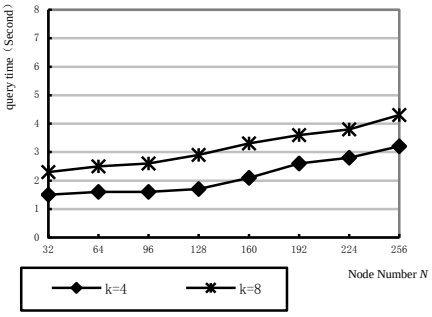


Figure 3 Average coverage maintenance costs



Figur 4 Average query time

stored. In two different conditions, the nodes number k=4 and k=8, the nodes in the network are randomly selected to generate 50 k-nearest neighbor query respectively. The simulation results are shown in Fig.3 and Fig.4.

Fig.3 describes the cover maintenance costs for social network cloud to take in the file storage, query (respectively set K = 4 and K = 8). It can be seen from Fig.3 that if K increases, the cover maintenance costs to store and search the target file will increase. This may be related to frequent cluster division and cluster merger when M is small.

Fig. 4 describes the query time for social network cloud to take in file query (respectively set K = 4 and K = 8). The query time refer to the average times from sending the query to returning all the results to meet the query condition. It can be seen from Fig.4 that if K increases, the query time will correspondingly increase. As can be also see that social network cloud has higher searching efficiency.

6. Conclusions and Future Work

In social network, traditional searching and locating method is faced with the characteristics of the high order correlation and implicit correlation for large-scale users, and user-association multidimensional and heterogeneity. This article analyze the main reason. This is because that the current studies mainly focus on the study of the correlation between social network and location information, the network topology structure of some user's characteristics, so as to improve the locating efficiency of social networks. They do not consider using the underlying network hierarchy structure and network protocols to solve the problems. This article introduces the peer-to-peer

distributed searching mechanism with the help of the cloud computing platform. This searching mechanism assigns user a logical identifier, and matches the underlying physical address and the upper users' logical address, so as to build the cloud logical topology structure of social network. This paper designs a *K* neighbor discovery algorithm. It is used to cluster the nodes according to the features of the user, so as to realize the quickly locating of social network. The performance of the algorithm is analyzed according to the user's searching logic path length of social network, and information amount of routing state. The simulation of the algorithm is evaluated by maintenance costs of average network aggregation coverage and query time. The simulation results demonstrate that social network cloud has good performance and searching efficiency. On the basis of the construction of a social network cloud, the further researches are about the semantic query, complex query *etc*.

Acknowledgments

This research was supported by Guangxi Natural Science Foundation (No.2013 GXNSFAA019337), Key project of Guangxi Education Department(No.2013ZD056), Guangxi Universities Key Project of Science and Technology Research (No.KY2015ZD099), Special Project of Yulin Normal University (No.2012YJZX04), Key Project of Yulin Normal University (No.2014YJZD05), Scientific Research Staring Foundation for the PHD Scholars of Yulin Normal University (No.G2014005), and Open Foundation for Guangxi Colleges and Universities Key Lab of Complex System Optimization and Big Data Processing(No. 2015CSOBDP0301).

*Corresponding author.
E-mail addresses: jgxyzxs@126.com(Xiaoshu Zhu), jgxyzj@126.com (Jie Zhang).

References

[1] Cranshaw J，Toch E，Hong Jetal. Bridging the gap between physical location and online social networks//Proceedings of the 12th ACM International Conference on Ubiquitious Computing. New York ，USA，2010:119-128
[2] Facebook Newsroom (2012): http://newsroom.fb.com/
[3] YouTube Global. (2012). Retrieved from http://youtubeglobal.blogspot.com/2012/01/ holy-nyans-60 -hours- perminute-and-4.html
[4] Han Yi, Xu Jin, Fang Bin-Xing, Zhou Bin, Jia Yan. Structural supportiveness theory on Social Networks. Chinese Journal of Computers，2014，37(4): 905-914
[5] Hung Chihchieh，Chang Chihwen，Peng Wenchih. Mining trafectory profiles for discovering iser communities//Proc of the 1st ACM SIGSPATIAL Int Workshop on Location Based Social Networks. New York: ACM，2009:1-8
[6] Deng Dong-Po，Chuang Tyng-Ruey，Lemmens Rob. Comceptualization of place via spatial clustering and cooccurrence analysis//Proc of the 1st ACM SIGSPATIAL International Workshop on Location Based Social Networks. New York: ACM，2009:49-56
[7] Zheng V W，Zheng Y，Yang Q. Joint learning user's activities and profiles from GPS data//Proc of the 1st ACM SIGSPATIAL Int Workshop on Location Based Social Networks. New York: ACM，2009:17-20
[8] Daisuki Yamamot—Itsu Takumi，Hiroshi Matsuo. Location-based social network services employing student cards for university//Proc of the 1st ACM SIGSPATIAL Int Workshop on Location Based Social Networks. New York: ACM，2009:21-24
[9] Karimi H A，Zimmerman B，Ozcelik A，*et al*. SonavNet: A framework for social navigation networks//Proc of the 1st ACM SIGSPATIAL Int Workshop on Location Based Social Networks. New York: ACM，2009:81-87
[10] Pultar E，Raubal M. A case for space: Physical and virtual location requirements in the couch Surfing social network// Proc of the 1st ACM SIGSPATIAL Int Workshop on Location Based Social Networks. New York: ACM，2009:88-91
[11] Doytsher Y，Galon B，Danza Y. Querying geo-social data by bridging spatial networks and social networks// Proc of the 2th ACM SIGSPATIAL Int Workshop on Location Based Social Networks. New York: ACM，2010:39-46

[12] Chow Chiyin，Bao Jie，Mokbel Mohamed F. Towads location-based social networking services// Proc of the 2th ACM SIGSPATIAL Int Workshop on Location Based Social Networks. New York: ACM，2010:31-38
[13] Ye Mao，Yin Peifeng，Lee Wangchien. Location recommendation for location-based social networks// Proc of the 18th SIGSPATIAL Int Conf on Advances in Geographic Information Systems. New York: ACM，2010:458-461
[14] Li Zhi, Zhang Hong, Liu Feng-Yu. Friend cluster based delay tolerant routing protocol in social Networks，Computer Science. 2012, 39（2）: 26~28+55
[15] Zhang Yu, Liu Yan-Bing, Xiong Gang, Jia Yan, Liu Ping, Guo Li. Survey on Succinct Representation of Graph Data. Journal of Software. 2014, 25(9):1937-1952.

Authors

Xiaoshu Zhu, She was born in Wuhan city, Hubei province, China. She received the master degree in computer science and technology from Guilin University of Electronic Technology, China in 2006. She is currently an professor at Yulin Normal University, Yulin, China. Her current main research interests include cloud storage, data mining, distributed network computing, *et al*.

Junhong Feng, She was born in Baoji city, Shaanxi province, China. She received the master degree in mining engineering from Xi'an University of Architecture and Technology, China in 2008. She is currently a lecturer at Yulin Normal University, Yulin, China. Her main research interests include expert system, data mining, *et al*.

Jie Zhang, He was born in Xianyang city, Shaanxi province, China. He received the Ph.D. degree in computer science and technology from Xidian University, China in 2013. He is currently an associate professor at Yulin Normal University, Yulin, China. His current main research interests include data mining, evolutionary computation, *et al*.