An Empirical Study on User Role Discovery Based on Clustering Algorithms and Optimizations in Location-Based Social Network

Ning Wang[1], Wenqing Zhu[1], Huiying Fang[1], Weimin Zhao[2]*

[1] School of Computer Science and Technology, Zhou Kou Normal University, China
[2] Henan Zhengda Tender Service Co., Ltd., China
wnpet@qq.com, 786632678@qq.com, 1378137855@qq.com, zhaoweimin@126.com

Abstract

Location-Based Social Network (LBSN) has been widely used in social lives. Role is an important concept in user's personalized analysis. Many automatic methods such as machine learning method and social network analysis method have been used in user role discovery in LBSN, however, the effectiveness of these methods has not been comprehensively analyzed. In this paper, firstly, the effectiveness of five clustering algorithms is comprehensively analyzed, including K-means algorithm, Bi-Kmeans algorithm, DBSCAN (Density-Based Spatial Clustering Application with Noise) algorithm, OPTICS (Ordering points to identify the clustering structure) algorithm and Agglomerate algorithms. Secondly, four strategies are designed to optimize the algorithm for user role discovery, namely GBK-means algorithm, RDK-means (Range and density k-means) algorithm, Canopy-based algorithm and reinforcement learning based algorithm. Thirdly, six data sets are used to validate the effectiveness of these algorithms, and the result shows that the optimization strategies are effective.

Keywords: Empirical evaluation, User role discovery, User role optimization, Canopy, Reinforcement learning

1 Introduction

With the development of wireless networks and location technology, it has become easier to determine and share the location information of individuals. Users prefer to share their activities or express their opinions according to their smart phones in the mobile network, such as writing blogs or chatting with friends. These activities are related to their geographical location, one of the most important aspects of human daily life, which founds Location-based Social Network (LBSN) [1].

Although these activities in LBSN seem fragmented and casual, they will exhibit a pattern in a large amount of data and long run. These patterns are shared by a set of users and reveal their life patterns and social characteristics, which can be abstracted as user roles [2].

However, user roles discovery in LBSN is challenging. As developers cannot communicate with users freely face-to-face, and users are not inclined to disclose their social related information in LBSN.

Massive user data has been accumulating in LBSN. How to analyze and abstract these data to reveal user roles and social characteristics and providing them with better services is an important task for mobile application developers. Therefore, clustering algorithms are used to discover user roles in this paper, and optimization strategies are proposed to improve their performance.

The contributions of this paper are as following:

Five clustering algorithms, namely K-means algorithm, DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm, Bi-Kmeans algorithm, OPTICS (Ordering points to identify the clustering structure) algorithm and Agglomerate algorithm are used to discover user roles based on user's check-in data, and the result are analyzed and discussed.

Four kinds of optimization strategy, namely GBK-means algorithm, RDK-means algorithm, Canopy-based algorithm and reinforcement learning based algorithm are proposed to improve the performance of user role discovery.

Based on six datasets, the performance of five clustering algorithms is summarized and discussed. And the effectiveness of the four optimization strategies is verified.

The paper is structured as follows:

Related works are described in section 2. User role discovery algorithms are detailed in section 3. Optimization strategies are detailed in section 4. Experiments are described in section 5. And finally, the paper is concluded, and the future works are summarized in section 6.

2 Related Work

Role discovery methods of general complex networks often focus on network structure, while LBSN are different from general complex networks. Users' attributes and behavioral characteristics are more attractive in LBSN. Therefore, identifying a group of similar and representative users is the core focus of role discovery in LBSN.

Role discovery methods in LBSN can be roughly divided into two parts. The first one is specific user role discovery [3-4], and the second one is nonspecific user role discovery [5-7]. Specific role discovery mainly focuses on specific scenarios where user roles and role characteristics are

*Corresponding Author: Ning Wang; E-mail: wnpet@qq.com
DOI: https://doi.org/10.70003/160792642024112506009

known. For example, in a virtual community, users can be divided into experts, opinion leaders and so on according to experience, specific role discovery method is suitable in this scenario. The methods used in specific role discovery mainly include classification and content analysis. While nonspecific user role discovery is suitable for complex or unknown scenarios, such as how to discover user roles in an unknown user group. The methods mainly include cluster analysis and statistical analysis.

2.1 Specific User Role Discovery

Specific role discovery belongs to targeted research. Discovery of the roles such as opinion leaders or experts belongs to this type of research.

Experts play an important role in social networking, and many methods have been proposed to discover it. Such as Graham et al. [8] used the method of digraphs to find out the experts who could best answer the questions in the forum. Forestier et al. [9] analyzes and reviews methods for identifying expert roles in social networks. Pournoor et al. [10] used the terms correlation matrix, vector space model and PageRank algorithm to propose a new hybrid model to identify experts. Neshati et al. [11] used the learning framework to analyze the four feature groups of topic similarity, emerging topics, user behavior and topic shift, and predicted the possibility of users becoming experts in the future.

Problems such as non-standard language and disordered resources exist in social networks. Opinion leaders are an important way to solve these problems. Therefore, it is necessary to identify opinion leaders.

Some researchers use the topic model to identify opinion leaders [4, 12-13]. Zhai [13] adopted an algorithm based on interest domain to analyze topic content with a combination of the structure of reply to relationship and user interest space in the network. And Song K combine multi-topic model with emotional analysis to identify the role of Opinion leader [12].

Other researchers use the characteristics of social networks such as outgoingness and point centrality to discover user roles [14-19]. Such as Y. Hu [14] using degree centrality, intermediate centrality and proximity centrality to discover opinion leaders.

Of course, specified roles include not only experts and opinion leaders, but also other roles such as influencer, power users, team members and so on, which will not be exhaustive here for brevity.

2.2 Nonspecific User Role Discovery

The nonspecific role discovery method aims to divide a group of users into several categories. This method strives to determine suitable roles for all users based on their characteristics and analyzes the meaning of roles and the differentiation between roles.

Social network analysis [20-22] and machine learning [23] are commonly used methods in nonspecific role discovery.

J. Fueller using the characteristics of external centrality, degree centrality and creative contribution in social network to discover roles such as creative users, passive reviewers [21]. Hacker [23] using machine learning method, namely K-means algorithm to identify roles in corporate social network.

Although methods are proposed to identify user roles, the performance of these methods has not been comprehensively analyzed. In this paper, the effectiveness of many clustering algorithms is summarized comprehensively, and four kinds of optimization strategies are used to improve the performance of the clustering.

3 User Rule Discovery Algorithms

Clustering can assist analysts to distinguish different groups from user data and summarize the activity patterns or habits of each group. As a module in data mining, it can discover some deep information distributed in data. In this chapter, many clustering algorithms are used to discover user roles.

3.1 Preliminaries

The vectorization of user data is the basis of clustering algorithms. In view of the fact that the data packet in LBSN contains context information such as time, latitude and longitude, as well as the view that analysts are interested in, the definition of user feature set are shown as follows.

The definitions used in the method are as follows:

Definition 1: Use Feature set (UFS)

UFS is the set of user features from different views of analyst under different contexts.

$$\mathrm{UFS} = \left\{ \mathrm{UF}^{\mathrm{ij}} \parallel 0 =< \mathrm{i} < \left|\mathrm{UCS}\right|, 0 =< \mathrm{j} < \left|\mathrm{VS}\right| \right\} \tag{1}$$

where:

$$UF^{\mathrm{ij}} = \begin{pmatrix} \mathrm{uf}^{\mathrm{ij}}{}_{11} & \cdots & \mathrm{uf}^{\mathrm{ij}}{}_{1\left|V^{j}\right|} \\ \vdots & \ddots & \vdots \\ \mathrm{uf}^{\mathrm{ij}}{}_{\left|UC^{i}\right|1} & \cdots & \mathrm{uf}^{\mathrm{ij}}{}_{\left|UC^{i}\right|\left|V^{j}\right|} \end{pmatrix} \tag{2}$$

In which UCS is the context set in the data, UC^i is a kind of context such as time, location and so on, and $UC^i \in$ UCS, VS is the view set that analysts used to observe user activity patterns according to their interests, V^j is a kind of view such as the category of point of interest, and $V^j \in$ VS. the label | | denotes the length of the set. UF^{ij} is a user feather matrix, the rows and columns of the matrix are context and view. User time-root category feature is a matrix with 24 rows and 9 columns, which is shown as Figure 1 in section 5.4.

3.2 User Role Discovery

With the complexity of software systems and the quantification of data, it is difficult to classify data accurately with only experience and professional knowledge, so the technology of multivariate analysis is introduced into numerical taxonomy, forming cluster analysis. Clustering can help analysts find the deep information hidden in the data,

distinguish different groups from a large number of data, identify the characteristics of each group, and summarize the patterns of these groups.

Five clustering algorithms are used to generate user roles from user check-in data, which are K-means algorithm, DBSCAN algorithm, Bi-Kmeans algorithm, OPTICS algorithm and Agglomerate algorithm.

- **K-means algorithm**
  K-means algorithm is a partition-based clustering algorithm. The algorithm selects k points as the initial clustering center and divides the data into K clusters and minimizes the distance between each sample and the center of its class. It has been widely used in practice because of its easy implementation and fast convergence speed. K-means algorithm is sensitive to the selection of the initial clustering center. In this article, we choice the points far away from others as the initial center points first, and then, the value of K is determined by calculating the root mean square error within the cluster. If the value decreases sharply before a certain value of k, and then, it slows down in the value of k, which forming a vivid "elbow", the K value is reasonable.
- **DBSCAN algorithm**
  DBSCAN algorithm does not need to specify the number of clusters, and the result is insensitive to data sequence. Two parameters are needed in this algorithm, namely search radius (denoted as eps), and minimum number of points in the range (denoted as minPts), a combination of these two parameters should be analyzed to ensure the effectiveness of the algorithm.
- **Bi-Kmeans algorithm**
  Bi-Kmeans algorithm is designed to solve the problem that K-means algorithm converges to the local minimum. The algorithm first takes all the points as a cluster, then divides the cluster into two parts, and then selects one of the clusters for further division. The division is repeated until the number of clusters specified by the user is obtained. The cluster selection is based on whether its partition can reduce SSE value to the greatest extent, in which SSE means sum of squared error.
- **OPTICS algorithm**
  OPTICS algorithm is an optimization of DBSCAN, which relaxes the search radius eps from a single value to a range value, therefore, the method is no longer sensitive to eps. As long as the value of minPts is determined, slight changes in eps will not affect the clustering results. OPTICS algorithm does not explicitly generate clusters, but instead generates an augmented cluster ranking. The ranking represents density-based clustering structure of each point. From this ranking, the clustering results of DBSCAN algorithm with any combination of the two parameters (eps, minPts) can be obtained.
- **Agglomerate algorithm**
  Agglomerate algorithm is a clustering method implemented through a bottom-up dependency tree. The core of the algorithm is the calculation of similarity distance.

According to different definitions of similarity distance, the algorithm includes three types: single link, complete link, and group average. Single link compares the minimum distance, complete link compares the maximum distance, and group average compares the average distance. Group average is used in this article.

4 User Role Optimization Strategies

Four kinds of optimization strategy are used to improve the user role discovery algorithm, namely, GBK-means algorithm, RDK-means algorithm, Canopy-based algorithm and reinforcement learning based algorithm.

4.1 GBK-means Algorithm

GBK-means [24] clustering algorithm is an improvement of the K-means algorithm based on bargaining games. The definitions used in the algorithms are shown as follows.

Definition 2: Data separation degree (SEP)

SEP represents the separation degree of data set S, which is calculated as formula 3. A small degree of separation means relatively centralized data, while large degree of separation means relatively dispersed data. In which MAXsi represents the maximum value of the i-th dimension data in data set S, and MINsi represents the minimum value.

$$SEP_s = \frac{\sum_{i=1}^{n} \frac{STP_{si}}{MAX_{si} - MIN_{si}}}{n} \tag{3}$$

Definition 3: Number of grid divisions (M)

M is used to divide each dimension of data set S into several parts, thus forming several grids, which is calculated as formula 4. In which k is the number of clusters.

$$M = \left\lfloor \frac{\sqrt{k}}{SEP_s} + 1 \right\rfloor \tag{4}$$

Definition 4: Dense grid (gd)

The data set S is projected into each grid, and the amount of data contained in each grid can be calculated. Dense grid denotes that the number of data contained in the grid is greater than or equal to the density threshold β. Where β is calculated as formula 5.

$$\beta = \frac{R}{M^n (1 - B)} \tag{5}$$

In which R is the total number of elements in the data set, B is the blank grid ratio, which is calculated in formula 6. Grids without data are called blank grids, and all blank grids form a blank grid set, labeled as G_B.

$$B = \frac{\|G_B\|}{M^n} \tag{6}$$

Based on the above definition, the process of GBK-means is shown in Algorithm 1.

Algorithm 1. GBK-means algorithm

Input: cluster number k, data set S with R elements and n dimensions

Output: k initial clustering centers

1: divide each dimension in S into M equal parts
2: divide the data set into M^n grids
3: identify the dense grid for each dimension, denoted as C_i
4: identify the data in C_i, denoted as D
5: initial the state for each of the dense grid C_i to unvisited
6: for each c in the dense grid set C:
7: if the state of c is unvisited:
8: denote the state of c to visited, assign a new cluster tag CT,
create list L, add c to L
9: else:
10: continue
11: endfor
12: gets L's header, check adjacent unvisited grids, change it to visited, if
it is dense grid, denote its current tag, and add it to the list L
13: connect to dense grid with the same tag, forming dense grid area
14: using k-means algorithm to obtain the primary cluster centers
15: calculate the average value of primary clusters as initial centers
16: if the number of initial centers j>k, DBSCAN algorithm is used to get k clustering centers
17: if j<=k, K-means++ algorithm is used to get clustering centers
18: print the resulting cluster centers

4.2 RDK-means

RDK-means [25] aims to improves the instability of k-means algorithm due to random generation of the centroids. The concept of global density is used in sample selection in the algorithm. And the sample points with the maximum density and distributed by a certain distance are selected as centroids. The definitions used in RDK-means are shown as follows.

Definition 5 Variance

Variance is used to measure the density of sample points, which is calculated in formula 7.

$$Var_i = \frac{1}{n-1}\Sigma\left(d\left(x_i, x_j\right) - m_i\right)^2 \tag{7}$$

In which $d(x_i, x_j)$ implies the distance between two points x_i and x_j. The label m_i denotes the average distance between the sample point and other points, which is calculated in formula 8.

$$m_i = \frac{1}{n}\sum\nolimits_{j=1}^{n} d\left(x_i, x_j\right) \tag{8}$$

Definition 6 Mean distance between samples

Mean distance between samples indicates the mean distance between each of the samples, which is calculated in formula 9.

$$r = \frac{2}{n(n+1)}\sum\nolimits_{i=1}^{n}\sum\nolimits_{j=1}^{i} d\left(x_i, x_j\right) \tag{9}$$

The process of RDK-means is shown in Algorithm 2.

Algorithm 2. RDK-means algorithm

Input: data set S with R elements and n dimensions, k clustering centroid

Output: k initial clustering centers

1: calculate the variance of all sampling points
2: add the minimum variance sample point to the centroid set
3: take the average distance of the minimum variance sampling points as the radius, and delete the sampling points in the circle
4: determine whether the initial centroid number is less than k. If yes, return to step 3; otherwise, go to the next step
5: save k initial centroid
6: calculate the distance between the centroid and save it in the distance list
7: take the minimum distance from the distance list, denoted as d_min
8: each centroid is plotted as a circle of radius γ
9: check whether the intersection of circles exists. if yes, go to step 10, else go to step 11
10: the points in the intersection area are divided into clusters at the center of the nearest intersection circle
11: the points are divided into clusters with the centers of the circles
12: delete the points
13: delete d_min from the distance list
14: check whether the distance list is empty; if yes, go to the next step otherwise, return to step 7
15: update the center point and take the mean of all sample points in the cluster as the new center point
16: check whether the center point still changes. If yes, return to step 6
17: print the final centroid

4.3 Canopy-based Algorithm

The main idea of Canopy algorithm [26] is to use a simple distance measure to divide all samples into many canopies, so as to classify a jumble of data into n data piles with certain rules. Although Canopy algorithm is not very accurate, it can be used to guide other clustering algorithms, such as k-means, to achieve better clustering results. Its advantages mainly include:

1. The algorithm can filter out the minority cluster and improve the anti-interference of the clustering algorithm.

2. Each center Point selected by the algorithm can assist the selection of other clustering algorithm centers.

A canopy-based algorithm is proposed to assist the selection of centroids of k-means algorithm. The detailed strategy is described in algorithm 3.

Algorithm 3. Canopy-based algorithm for clustering centers selection

Input: user feature set UFS, T_1,T_2
Output: a set of clustering centers
1: construct a list L to store the feature set for each user
2: initialize T_1 and T_2
3: initialize canopy set canopy_set=null
4: do:
5: fetch f_i in UFS to construct a canopy, and denoted as $canopy_{[i]}$
6: remove f_i from UFS
7: canopy_set.add($canopy_{[i]}$)
8: fetch a feature f_k
9: flag_in=0
10: for each c in canopy_set:
11: calculate f_k.distance(c), and denoted as d_{kc}
12: if $d_{kc} <= T_1$:
13: c.add(f_k)
14: flag_in=1
15: if flag_in == 0:
16: using f_k to construct a canopy, denoted as $canopy\_f_k$
17: canopy_set.add($canopy\_f_k$)
18: for each c in canopy_set:
19: calculate f_k.distance(c), and denoted as d_{kc}
20: if $d_{kc} <= T_2$:
21: remove f_k from list L
22: while list L is not empty
23: return canopy_set

4.4 Reinforcement Learning based Algorithm

The idea of reinforcement learning is optimizing the agent's behaviors to get a good benefit in a complex environment [27-29]. As we know that the optimal strategy in each step may not necessarily yield the best long-term benefits in a complex environment, therefore, the benefit consists of two parts, namely the instant reward and long-term value. The first one is defined to measure the benefit of a behavior in the current environment, the larger it is meaning better the behavior is. The second one is defined to measure the benefit of a sequence of behavior in a long term.

In this paper, reinforcement learning is used optimize user role discovery. The definitions and evaluation metrics used in the algorithm is shown as follows.

Definition 7: State set S

S denotes all the possible states in the attempt process. Which consist of a set of states.

Definition 8: Behavior set B.

B denotes all the behaviors that may appear in the learning process. Which consists of a set of behaviors.

Definition 9: State behavior set B_s.

B_s denotes the possible behaviors set in current state s. Note that B_s is a subset of B.

Definition 10: Instant reward $Reward_s^b$

$Reward_s^b$ denotes the reward obtained when the agent adopts behavior b in current state s.

Definition 11: Long-term value V

V denotes the long terms benefits obtained when the agent executes a sequence of behaviors.

Silhouette coefficient is an important metric in clustering evaluation [30], the calculation of it is shown in formula 10:

$$Se(UF_i) = \frac{ORMin(UF_i) - TRMax(UF_i)}{Max(ORMin(UF_i), TRMax(UF_i)} \quad (10)$$

In the formula, TRMax (UF_i) denotes the max distance between UF_i and other user features in the same role. ORMin (UF_i) denotes the min distance between UF_i and the centroid of other roles.

Se(UF_i) is used to measure the instant reward of a single activity, that is, the activity that assigns a user to a role. To measure the long-term value of the algorithm, average contour coefficient is used, which is shown in formula 11. In which m is the number of users.

$$Se(UF) = \frac{1}{m}\sum_{i=1}^{m} Se(UF_i) \quad (11)$$

The state of the algorithm is a matrix with |U| rows and |R| columns, where |U| is the number of users and |R| is the number of roles. The initialization of the state is based on the results of a clustering algorithm, the update rule of the state is shown in formula 12, in which β is a parameter in [0,1], used to adjust the impact of the instant reward obtained in one assignment, The larger β is, the greater the impact of instant reward is on the status.

$$S^{n+1}[i\text{-}1][j\text{-}1] = (1\text{-}\beta) * S^n[i\text{-}1][j\text{-}1] + \beta * Se(UF_i)_j \quad (12)$$

Based on the definitions and evaluation metrics introduced above, the method is described in algorithm 4.

Algorithm 4. Reinforcement learning based algorithm for user role discovery optimization

Input: user set U; Role set R
Output: correspondence between users and roles
1: for each u in U /* State initialization based on the clustering result*/
2: for each r in R
3: S[positon_u][position_r] = α
4: endfor
5: endfor
6: for each u in U
7: for each r in R
8: attempt to assign u to the role r
9: calculate the instant reward of the assignment according to formula 7
10: endfor
10: take the maximum value of the instant reward, and the correspond r
11: assign u to r
12: update the centroid of r
13: U=U-u
14: update the state S according to formula 9
15: endfor
16: calculate the long-term value V of the algorithm according to formula 8

17: if V $< \gamma$
18: return to line 6 to continue the optimization process
19: else
20: end of the algorithm
21: return V and final state of S, namely the correspondence between users and roles

5 Experiments

Experiments are conducted to verify the effectiveness of these methods. To verify that the method is not data sensitive, 6 data sets are used in the experiment.

5.1 Data Set

The data set used in [31] is adopted in this paper as data set 1. The data set used in [32] contains check-in data in hundreds of countries and cities. In this paper, data set 2 to data set 6 are generated from users in different countries randomly, corresponding to the country of American, Malaysia, Indonesia, Mexico and Japan. The user number in each country is two thousand, as the user in some countries are limited. The user number, point of interest (POI) number and check-in number of these data sets are shown in Table 1.

Table 1. Brief information of the data sets

	User number	POI number	Check-in number
Data set 1	1083	38333	227428
Data set 2	2000	220271	795024
Data set 3	2000	50514	324100
Data set 4	2000	57507	756598
Data set 5	2000	38396	397114
Data set 6	2000	79437	818735

The data items in these data sets mainly includes the user number, POI number, POI category number, POI category name, latitude and longitude of the check-in activity and time of the check-in activity.

The data set used in this article can be found in URL: https://pan.baidu.com/s/13zYwoV_jGjA4QwtqHedETg, and the extracted code is "asdf".

5.2 Research Questions

Based on the experimental objectives, the research questions of the experiment are as follows:

RQ1: Are these clustering algorithms effective to discover user roles from user data?

RQ2: What is the performance of these clustering algorithms in discovering user roles?

RQ3: Are these optimization strategies used in this paper effective in user role discovery optimization?

5.3 Evaluation Metrics

The evaluation metrics used in the experiment include mean silhouette coefficient, Calinski-Harabasz (C-H) score, Davies-Bouldin (D-B) score.

- **Mean silhouette coefficient**

Silhouette coefficient can be used to evaluate the effect of clustering. Its value range is between -1 and 1, the larger the value is, the better the clustering effect is, which means the distance between the points in current cluster is small, while the distance between current cluster and nearby cluster is large. Mean silhouette coefficient is used to measure the overall clustering effect. The calculation of silhouette coefficient and mean silhouette coefficient are shown in formula 7 and formula 8.

- C-H score

C-H score is calculated by evaluating inter cluster variance and intra cluster variance. The calculation is shown in formula 13.

$$s = \frac{\frac{SS_B}{k-1}}{\frac{SS_W}{N-k}} \tag{13}$$

In which k represents the number of cluster categories, N represents the number of data, SS_B represents inter cluster variance, and SS_W represents intra cluster variance. The calculation of SS_B and SS_W are shown in formula 14 and formula 15.

$$SS_B = \mathrm{tr}\left(B_k\right) \tag{14}$$

$$SS_W = \mathrm{tr}\left(W_k\right) \tag{15}$$

In which the calculation of B_k and W_k are shown in formula 16 and formula 17.

$$B_k = \sum_{q=1}^{k} n_q \left(c_q - c_E\right) \left(c_q - c_E\right)^T \tag{16}$$

$$W_k = \sum_{q=1}^{k} \sum_{x \in C_q} n_q \left(x - c_q\right) \left(x - c_q\right)^T \tag{17}$$

In which n_q is the total number of points in cluster q, c_q is the centroid of cluster q, c_E is the centroid of all points in the data and C_q is the set of data in cluster q.

- D-B score

Calculate the sum of the diameter for two clusters, divide by the distance between the centroid of these two clusters, finally, the maximum value is taking as D-B score. Smaller the D-B score is means smaller the intra class distance is and greater the inter class distance is, which means better the clustering performance is. The calculation of D-B score is shown in formula 18 and formula 19.

$$R_{ij} = \frac{s_i + s_j}{d_{ij}} \tag{18}$$

$$\mathrm{DB\_score} = \frac{1}{k} \sum_{i=1}^{k} max_{i \neq j} R_{ij} \tag{19}$$

In which s_i and s_j are the diameters for the two clusters, and d_{ij} is the distance between the centroid of these two clusters.

5.4 Results and Analysis

User feature matrix is the basis of user role discovery, based on the data items in the data set, user feature matrix is constructed based on time and root category, time is segmented in hours in one day, and root category has 9 values, therefore, user feature matrix is a matrix with 24 rows and 9 columns. An example of a user feature is shown in Figure 1, in which x axis denotes time, y axis denotes root category, and z axis denotes the total check-in times of a user in corresponding time and root category.

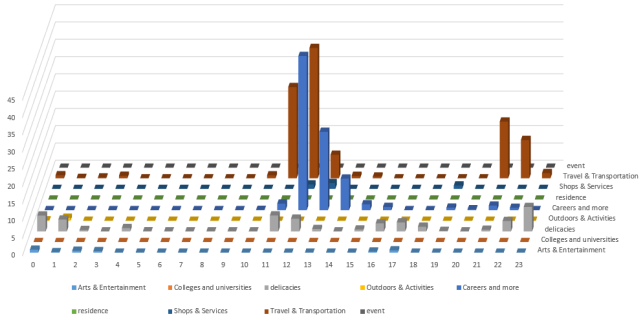

Figure 1. Illustration of user feature matrix

- RQ1: Are these clustering algorithms effective to discover user roles from user data?

The results show that the different categories in the clustering results are effective to summarize the habits and customs in daily life of a group of users. Thereby providing a feasible means to discover user social roles. The schematic of a set of user roles discovered through DBSCAN clustering algorithm is shown in Figure 2 to Figure 6.

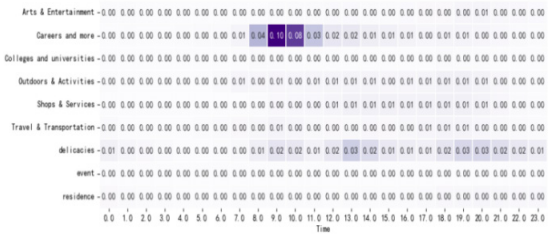

Figure 2. Stable worker

As shown in Figure 2, around 08:00 to 12:00 a.m., users in this role have obvious frequent activities in the category of careers. Which means that users in this role have a stable job and should be at work in the morning. We refer to users in this category as stable worker role.

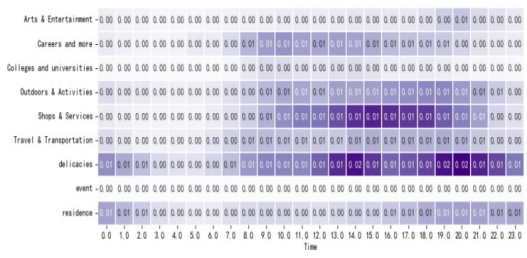

Figure 3. Leisure urban residents

As shown in Figure 3, from approximately 12:00 o'clock, user activities are beginning to increase significantly, and until around 22:00 o'clock, user activities are reduced significantly. The activities of these users mainly including shops and service, delicious food, some outdoors activities and travel & transportation activities. Which indicates that users in this role enjoy activities such as food and shopping in the mall, they may often take public transport to go home, enjoy a free and leisurely life as they like, they have a regular schedule and refuse to stay up late. We refer to users in this category as leisure urban residents' role.

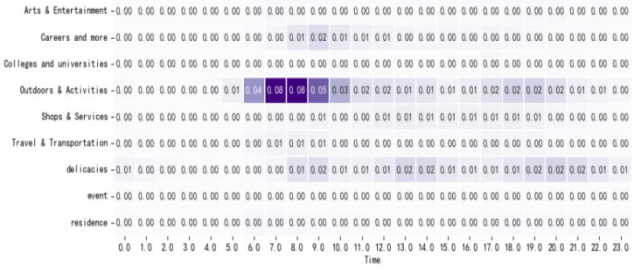

Figure 4. Morning exercise enthusiasts

As shown in Figure 4, the significant characteristics of users in this cluster is they have many activities in outdoors & activities category from about 6:00 o'clock to 9:00 o'clock. This indicates that users in this role like to enjoy exercising such as running in the morning. We refer to users in this category as morning exercise enthusiast's role.

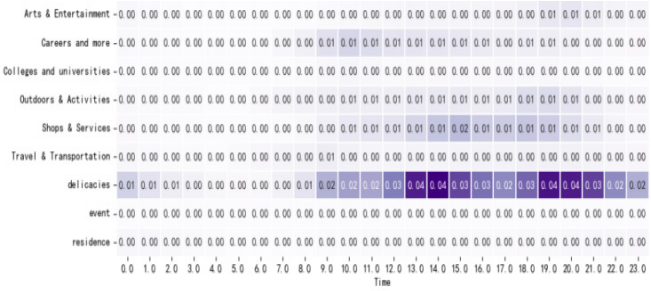

Figure 5. Health preserving gourmet

As shown in Figure 5, users in this cluster have many activities in food category, and the time distribution is around 12:00 noon to 22:00 pm, indicating that users in this cluster like to enjoy fine food, and they don't tend to stay up late. We refer to users in this category as health preserving gourmet role.

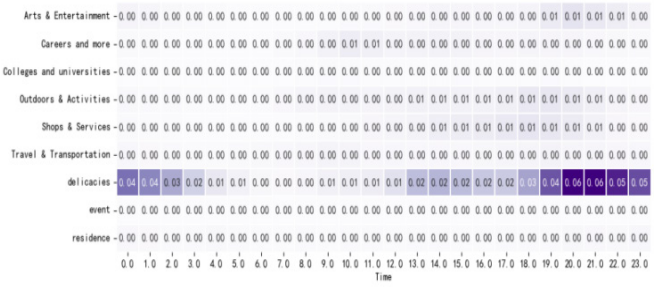

Figure 6. Night owl

As shown in Figure 6, users in this cluster also have many activities in food category, they began to become noticeably active after dark, that is to say, from around 18:00 to 19:00,

They remain active after 12 pm and gradually subside after 2:00 to 3:00 am. Which indicates that users in this cluster indulge in nightlife, they may be young people who enjoy beer and barbecue. We refer to users in this category as the night owl role.

After the clustering result is identified, we calculated the proportion of user group, the detailed information is shown in Figure 7.

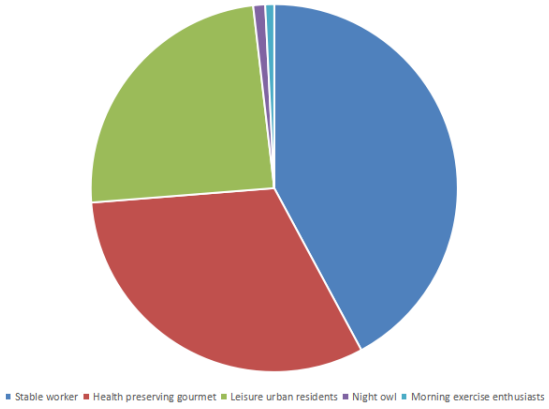

Figure 7. Proportion of user group for each clustering category

As shown in the figure, stable workers have the most users in the category, followed by health preserving gourmet and leisure urban residents. These three categories have a large number of users, accounting for over 95% of all users. The night owl and morning exercise enthusiast's category have little users.

As the user characteristics in social network cannot be exhaust, we cannot ensure the completeness of the clustering result, however, as the clustering algorithms used in this article can discover user roles effectively, we think it is helpful for data manager to understand user features.

- RQ2: What is the performance of these clustering algorithms in discovering user roles?

Based on the evaluation metrics defined in section 5.3, performance of these clustering algorithms used in this paper are verified based on the data set defined in section 5.1.

The performance of these algorithms on mean silhouette coefficient are shown in Figure 8.

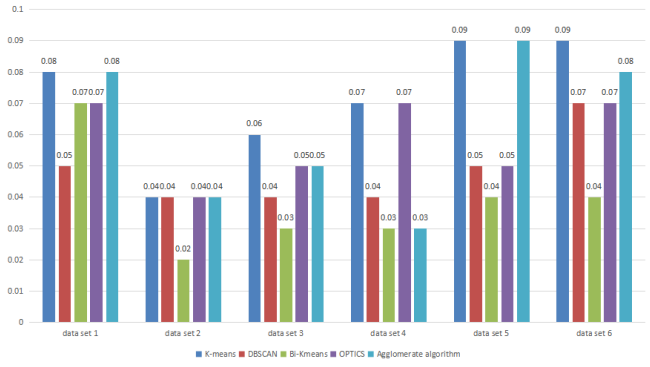

Figure 8. Mean silhouette coefficient of these algorithms

From the figure we find that K-means algorithm has the highest value in 6 datasets, although in 4 of the data sets, it's value is the same as some of the other algorithms such as DBSCAN and agglomerate algorithm. The agglomerate algorithm has the highest value in 3 data sets, namely data set 1, data set 2 and data set 5. OPTICS algorithm has the highest value in 2 data sets, namely data set 2 and data set 4. DBSCAN algorithm has the highest value in 1 data set, namely data set 2. And Bi-Kmeans algorithms do not have the highest value in all of the 6 data sets.

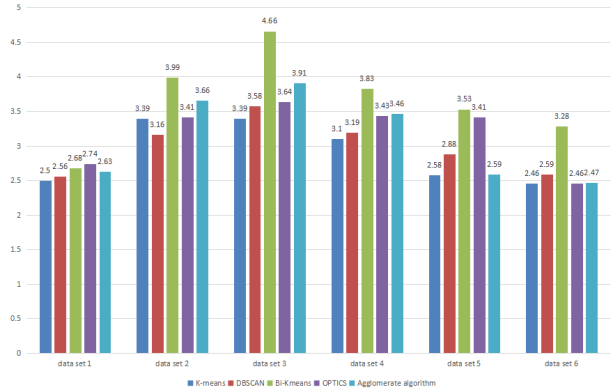

Figure 9. D-B score of these algorithms

The smaller the value of D-B score is the better the clustering effect is. As shown in Figure 9, K-means algorithm has the smallest value in five data sets except in data set 2. DBSCAN algorithm has the smallest value in one data set, namely data set 2. The Bi-Kmeans algorithm has the biggest value in five data sets except in data set 1. OPTICS algorithm has the biggest value in one data set, namely data set 1.

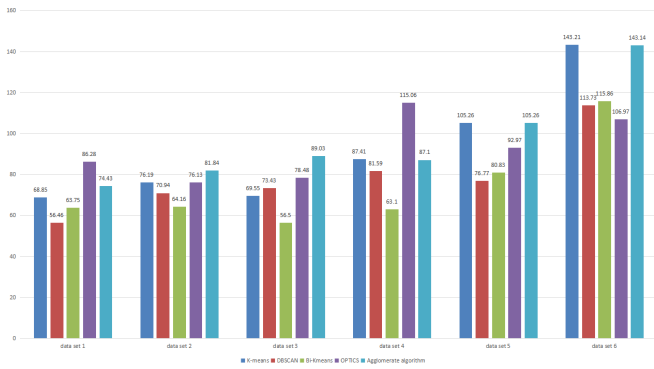

Figure 10. C-H score of these algorithms

The bigger the value of C-H score is the better the clustering effect is. As shown in Figure 10, K-means algorithm has the biggest value in two data sets, namely data set 5 and data set 6. The OPTICS algorithm has the biggest value in two data sets, namely data set 1 and data set 4. The agglomerate algorithm has the highest value in 2 data sets, namely data set 2 and data set 3. The Bi-Kmeans algorithm has the smallest value in three data sets, namely data set 2, data set 3 and data set 4. DBSCAN algorithm has the smallest value in two data sets, namely data set 1 and data set 5. OPTICS algorithm has the smallest value in one data set, namely data set 6.

From the results, we find that no algorithm outperforms all other algorithms in all of the datasets and evaluation metrics. Therefore, several rules are defined to evaluate the performance of these algorithms comprehensively:

(1) If the algorithm performs best in one data set and one evaluation metric, its value is increased by one.

(2) If the algorithm performs not the best and not the worst in one data set and one evaluation metric, its value remains unchanged.

(3) If the algorithm performs worst in one data set and one evaluation metric, its value is decreased by one.

Based on the rules defined, the performance of these algorithms is summarized, which is shown in Figure 11.

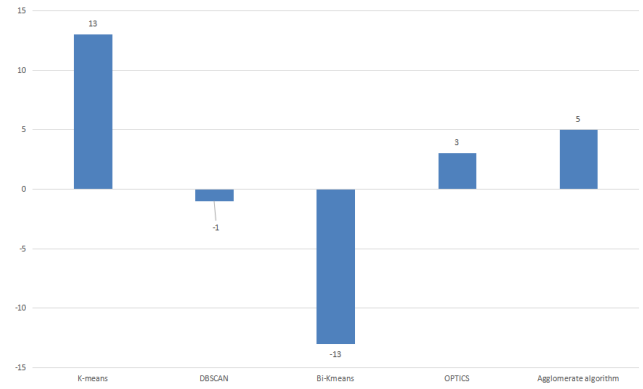

Figure 11. Comprehensive performance of these algorithms

As shown in the figure, K-means algorithm performs the best, agglomerate algorithm performs the second, followed by OPTICS algorithm and DBSCAN algorithm, and Bi-Kmeans performs the worst.

- RQ3: Are these optimization strategies used in this paper effective in user role discovery optimization?

As analyzed in RQ2, K-means algorithm outperforms other algorithms, therefore, it is selected to compare with the four optimization strategies, namely GBK-means algorithm, RDK-means algorithm, Canopy-based algorithm-based algorithm.

The value of mean silhouette coefficient for these optimization strategies is shown in Figure 12.

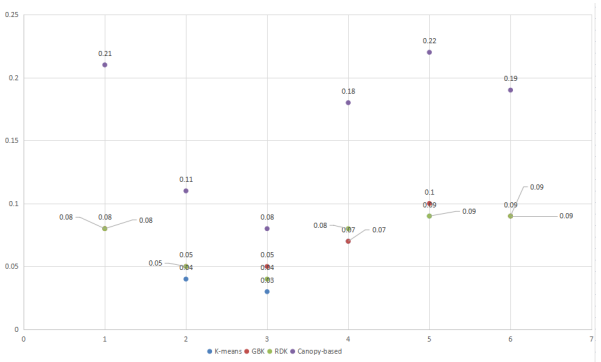

Figure 12. Mean silhouette coefficient for optimization strategies

As shown in the figure, mean silhouette coefficient values for the three optimization strategies outperform or not less than K-means algorithm in the six data sets. The values of GBK and RKD algorithm are slightly higher than the value of K-means algorithm, and the value of Canopy-based algorithm is much larger than the value of K-means.

The value of C-H score for these optimization strategies is shown in Figure 13.

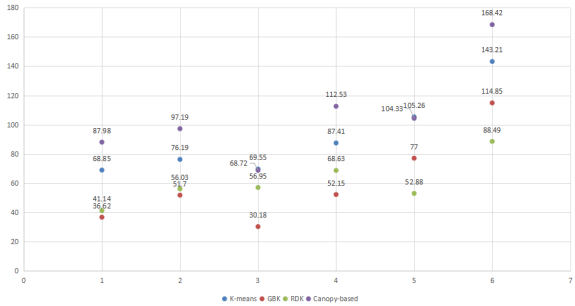

Figure 13. C-H score for optimization strategies

As shown in the figure, C-H score values for the Canopy-based algorithm are larger than the value of K-means in four of the data sets, namely data set 1, data set 2, data set 4 and data set 6, and are slightly smaller in the other two of the data sets. However, the C-H score of the other two optimization strategies namely GBK and RDK algorithms are smaller than the value of K-means in the six data sets.

The value of D-B score for these optimization strategies is shown in Figure 14.

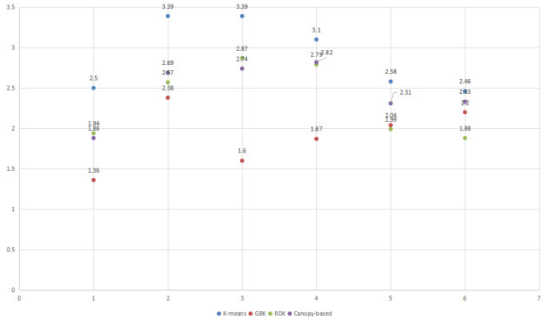

Figure 14. D-B score for optimization strategies

As shown in the figure, D-B score values for the three optimization strategies are smaller than the value of K-means in the six data sets, in which the values of GBK algorithm are the smallest in four data sets, namely data set 1, data set 2, data set 3 and data set 4, and the values of RDK algorithm are the smallest in two data sets, namely data set 5 and data set 6.

As shown and discussed above, no optimization strategy outperforms others in all of the datasets and evaluation metrics. Therefore, based on the rules defined in RQ2, the performance of these optimization strategies is evaluated comprehensively. The result is shown in Figure 15.

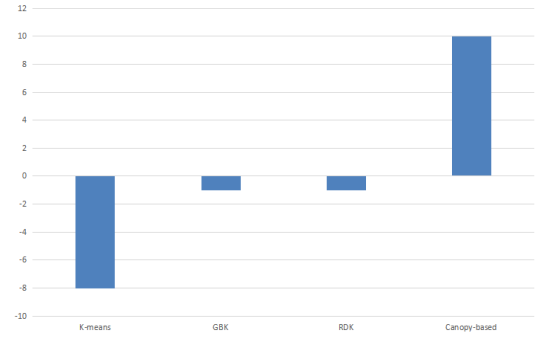

Figure 15. Comprehensive performance of optimization strategies

As shown in the figure, Canopy-based method performs the best, GBK and RDK algorithm performs the second, and K-means performs the worst.

The effectiveness of reinforcement learning has not been compared and discussed with the other four algorithms. Because the parameters for the convergence of reinforcement learning algorithms can be manually set. In the experiments, we set the convergence parameters of reinforcement learning to the best performance of other algorithms, the reinforcement learning algorithm can effectively converge ultimately, however, the running time of algorithms varies in different datasets.

5.5 Validity Threats

"The validity threats of the experiment mainly including conclusion validity, internal validity and external validity. To ensure validity of the conclusion validity, the cluster and optimization algorithms, research questions of the experiments, data set of the experiments and evaluation metrics of the experiments are carefully designed. For internal validity, the results of the experiments show that clustering algorithms can discover user roles, and the optimization strategies are effectively. For external validity, the data sets used in this paper are publicly available, which reduces the external validity."

6 Conclusion and Future Work

It is challenging for mobile application developers to communicate with users directly. Discovering user roles based on the analysis of user data can assist developers to understand user features and requirements. In this study, several clustering algorithms are used to discover user roles based on check-in data, and four kinds of optimization strategies are used to improve the effectiveness of these clustering. Experiments are conducted to verify these of these algorithms, and results show that they are effective to discover user roles and improve the performance of the clustering.

Future work includes the following parties. Firstly, combine the check-in data used in this paper with other types of data such as comments data and twitter data to discover user roles comprehensively. The effectiveness of the optimization strategies should be verified in larger data sets. Thirdly, efficiency of the algorithms should also be taken into consideration in the future work.

Acknowledgment

This research was funded by National Natural Science Foundation of China, grant number No. U1504602 and No. 61902447; this research was funded by High-level Talent Scientific Research Start-up Fund under Grant No. ZKNUC2021023.

References

[1] Z. Yu, Location-based social networks: users, in: Y. Zheng, X. Zhou (Eds.), *Computing with Spatial Trajectories*, Springer, New York, 2011, pp 243-276.

[2] J. V. Hacker, F. Bodendorf, P. Lorenz, A framework to identify knowledge actor roles in enterprise social networks, *Journal of Knowledge Management*, Vol. 21, No. 4, pp. 817-838, August, 2017.

[3] P. K. Roy, J. P. Singh, A. Nag. Finding Active Expert Users for Question Routing in Community Question Answering Sites, Proceedings of the Machine Learning and Data Mining in Pattern Recognition, New York, NY, USA, 2018, pp. 440-451.

[4] T. Zhao, H. Huang, X. Fu, Identifying Topical Opinion Leaders in Social Community Question Answering, *Proceedings of the Database Systems for Advanced Applications*, Gold Coast, QLD, Australia, 2018, pp. 372-387.

[5] A. Mccallum, X. Wang, A. Corrada-Emmanuel, Andres Corrada-Emmanuel. Topic and role discovery in social networks with experiments on enron and academic email, *Journal of Artificial Intelligence Research*, Vol. 30, pp. 249-272, October, 2007.

[6] A. Peleshchyshyn, V. Vus, O. Markovets, S. Albota, Identifying Specific Roles of Users of Social Networks and Their Influence Methods, *2018 IEEE 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies*, Lviv, Ukraine, 2018, pp. 39-42.

[7] Y.-H. Chen, X.-M. Yang, Research on the Classification of Users in Open Knowledge Community-A Case Study of Chinese Wikipedia, *Modern Educational Technology*, Vol. 26, No. 6, pp. 47-53, June, 2016.

[8] M. W. Graham, E. J. Avery, S. Park, The Role of social media in Local Government Crisis Communications, *Public Relations Review*, Vol. 41, No. 3, pp. 386-394, September, 2015.

[9] M. Forestier, A. Stavrianou, J. Velcin, D.-A. Zighed, Roles in Social Networks: Methodologies and Research Issues, *Web Intelligence and Agent Systems: An international Journal*, Vol. 10, No. 1, pp. 117-133, 2012.

[10] E. Pournoor, J. Rezaenoor, A New Expert Finding Model Based on Term Correlation Matrix, *Iranian Journal of Information Processing & Management*, Vol. 30, No. 4, pp. 1147-1171, September, 2015.

[11] M. Neshati, Z. Fallahnejad, H. Beigy, On Dynamicity of Expert Finding in Community Question Answering, *Information Processing & Management*, Vol. 53, No. 5, pp. 1026-1042, September, 2017.

[12] K. Song, D. Wang, S. Feng, D. Wang, G. Yu, Detecting Positive Opinion Leader Group from Forum, *Proceedings of the Web-Age Information Management*, Harbin, China, 2012, pp. 95-101.

[13] Z. Zhai, H. Xu, P. Jia, Identifying Opinion Leaders in BBS, *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, Sydney, NSW, Australia, 2008, pp. 398-401.

[14] Y. Hu, Identification of Opinion Leaders in the Network Academic Community, *Journal of Library and Information Science*, Vol. 2, No. 11, pp. 72-77, 2017.

[15] F. Probst, L. Grosswiele, R. Pfleger, Who will lead and who will follow: Identifying Influential Users in Online Social Networks, *Business & Information Systems Engineering: The International Journal of WIRTSCHAFTS INFORMATIK*, Vol. 5, No. 3, pp. 179-193, 2013.

[16] I. Kayes, X. Qian, J. Skvoretz, A. Iamnitchi, How Influential Are You: Detecting Influential Bloggers in a Blogging Community, *4th International Conference on Social Informatics*, Lausanne, Switzerland, 2012, pp. 29-42.

[17] X. Wu, J. Wang, Micro-blog in China: Identify influential users and automatically classify posts on Sina micro-blog, *Journal of Ambient Intelligence & Humanized Computing*, Vol. 5, No. 1, pp. 51-63, February, 2014.

[18] Y. Zhang, X. Li, T. Wang, Identifying Influencers in Online Social Networks: The Role of Tie Strength, *International Journal of Intelligent Information Technologies*, Vol. 9, No. 1, pp. 1-20, January-March, 2013.

[19] S. Chatterjee, R. Chaudhuri, D. Vrontis, R. Piccolo, Enterprise social network for knowledge sharing in MNCs: Examining the role of knowledge contributors and knowledge seekers for cross-country collaboration, *Journal of International Management*, Vol. 27, No. 1, pp. 1-14, March, 2021.

[20] S.-L. Toral, M.-R. Martínez-Torres, F. Barrero, Analysis of Virtual Communities Supporting OSS Projects Using Social Network Analysis, *Information and Software Technology*, Vol. 52, No. 3, pp. 296-303, March, 2010.

[21] J. Fuller, K. Hutter, J. Hautz, K. Matzler, User Roles and Contributions in Innovation-Contest Communities, *Journal of Management Information Systems*, Vol. 31, No.1, pp. 273-307, Summer, 2014.

[22] O. Ma, X. Luo, M. Zhao, User Behaviour Network Based User Role Mining of Web Event, *2018 14th International Conference on Semantics, Knowledge and Grids*, Guangzhou, China, 2018, pp. 211-217.

[23] J. Hacker, K. Riemer, Identification of User Roles in Enterprise Social Networks: Method Development and Application, *Business & Information Systems Engineering*, Vol. 63, No. 4, pp. 367-387, August, 2021.

[24] M.-J. Rezaee, M. Eshkevari, M. Saberi, O. Hussain, GBK-means clustering algorithm: An improvement to the K-means algorithm based on the bargaining game, *Knowledge-Based Systems*, Vol. 213, pp. 1-13, February, 2021.

[25] Q. Zhang, Z. Wu, Y. Kamiya, B. Wang, Lift-down trajectory generation for multi-joint robot based on repeatedly direct kinematics algorithm, *2010 IEEE International Conference on Mechatronics & Automation*, Xi'an, China, 2010, pp. 65-70.

[26] G. Zhou, Improved Optimization of Canopy-Kmeans Clustering Algorithm Based on Hadoop Platform, *The International Conference on Information Technology and Electrical Engineering*, Xiamen, Fujian, China, 2018, pp. 1-6.

[27] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, D. Hassabis, Human-level control through deep reinforcement learning, *Nature*, Vol. 518, No. 7540, pp. 529-533, February, 2015.

[28] Y. Cheng, D. Li, W.-E. Wong, M. Zhao, D. Mo, Multi-UAV Collaborative Path Planning using Hierarchical Reinforcement Learning and Simulated Annealing, *International Journal of Performability Engineering*, Vol. 18, No. 7, pp. 463-474, July, 2022.

[29] S. Mishra, A. Arora, Double Deep Q Network with Huber Reward Function for Cart-Pole Balancing Problem, *International Journal of Performability Engineering*, Vol. 18, No. 9, pp. 644-653, September, 2022.

[30] A. O. R. Rodriguez, M. A. Riaño, P. A. G. García, C. E. M. Marín, Advanced Clustering Techniques for Emotional Grouping in Learning Environments Using an AR-Sandbox, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, Vol. 30, No. 3, pp. 427-442, June, 2022.

[31] D. Yang, D. Zhang, V.-W. Zheng, Z. Yu, Modeling user activity preference by leveraging user spatial temporal characteristics in LBSNs, *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, Vol. 45, No. 1, pp. 129-142, January, 2015.

[32] D. Yang, D. Zhang, B. Qu, Participatory cultural mapping based on collective behavior data in location-based social networks, *ACM Transactions on Intelligent Systems and Technology*, Vol. 7, No. 3, pp. 1-23, April, 2016.

Biographies

Ning Wang received the M.E. degree from Northwestern Polytechnical University in 2009. Currently, he is worked in the School of Computer Science and Technology at Zhoukou Normal University. His research interests include data analysis and artificial intelligence.

Wenqing Zhu is now a bachelor candidate in Zhoukou Normal University in China. Currently. He participated in the "Lanqiao Cup" competition and the "Internet+" competition, and had obtained a silver award. His research interests include requirement engineering and social network.

Huiying Fang is studying in Computer Science and Technology from Zhoukou Normal University of China. Her research interests include recommendation algorithm and artificial intelligence. Her current research lies at complete data analysis tasks using deep learning.

Weimin Zhao graduated from Zhengzhou University in 2009. Currently, she is working at Henan Zhengda Tender Service Co., LTD. Her position is project management and her research direction is information engineering.