RESEARCH Open Access

# A user interest community evolution model based on subgraph matching for social networking in mobile edge computing environments



Liang Jiang<sup>1,2</sup>, Lu Liu<sup>3\*</sup>, Jingjing Yao<sup>4</sup> and Leilei Shi<sup>1,2</sup>

### Abstract

With the rapid development of mobile edge computing, mobile social networks are gradually infiltrating into our daily lives, in which the communities are an important part of social networks. Internet of People such as online social networks is the next frontier for the Internet of Things. The combination of social networking and mobile edge computing has an important application value and is the development trend of future networks. However, how to detect evolutionary communities accurately and efficiently in dynamic heterogeneous social networks remains a fundamental problem. In this paper, a novel User Interest Community Evolution (UICE) model based on subgraph matching is proposed for accurately detecting the corresponding communities in the evolution of the user interest community. The community evolutionary events can be quickly captured including forming, dissolving, evolving and so on with the introduction of core subgraph. A variant of subgraph matching, called Subgraph Matching with Dynamic Weight (SMDW), is proposed to solve the problem of updating the core subgraph due to the change of core user's interest when tracking evolutionary communities. Finally, the experiments based on the real datasets have been designed to evaluate the performance of the proposed model by comparing it with the state-of-art methods in this area and complete data processing through the local edge computing layer. The experimental results demonstrate that the UICE model presented in this paper has achieved better accuracy, higher efficiency and better scalability against existing methods.

Keywords: Edge computing, Dynamic heterogeneous social networks, UICE, SMDW

### Introduction

With the rapid development of mobile edge computing, mobile social networks play an increasingly important role in people's daily lives. Mobile edge computing emerged as a new paradigm application, pushing the frontiers of computing applications, data and services from centralized nodes to the mobile edge of the network, forming a useful supplement to cloud computing, and obtaining a better user experience through resource collaboration [1]. The combination of social networking

and mobile edge computing has an important application value and is the development trend of future networks [2]. In order to optimize the benefits of the network, there are still many important challenges that need to be solved urgently, among which social network data analysis is one of them [3]. Many advanced machine learning technologies, for example, deep learning method, have achieved great success in social media data analysis [4]. The research of community tracking and other analysis in dynamic heterogeneous social networks is attracting worldwide attention recently [5–7]. Scholars in many fields analyze online social networks from different perspectives and detect communities in the

<sup>3</sup>School of Informatics, University of Leicester, Leicester, UK Full list of author information is available at the end of the article



<sup>\*</sup> Correspondence: I.liu@leicester.ac.uk

network to analyze the whole network [8, 9]. It is one of the methods that people always pay attention to, and it is also a hot and difficult point in the field of research. In social networks, nodes generally represent a single individual, while edges represent relationships that exist between individuals, such as classmate relationships, kinship relationships, and friend relationships [10, 11], while communities represent the circle of friends with common interests or other common attributes [12]. The significance of interest community evolution analysis is not only to explore the existing evolution rules of the community, but also to predict the future evolution path according to the existing information [13]. For example, in the e-commerce platform, personalized recommendation can be accurately made through the evolution analysis of user interest, so as to stimulate user consumption and improve product sales. And in the process of dissemination of online public opinion (topics of interest to users), by analyzing the content and dissemination rules of existing topic communities, predicting the dissemination trend of negative public opinion and mining the core nodes of leading public opinion, timely measures are taken to control public opinion and reduce the adverse effects of public opinion.

Early research on the community focused on community discovery of static networks, and many algorithms emerged [14, 15] as well. In recent years, community research has been focused on the process of community structure change in dynamic networks. Through the study of the dynamics and evolution process of the community in online social networks, it is of great help to better understand the internal structure of online social networks [16, 17], which is a network structure formed by many users based on their diversified interests. In order to investigate the evolution of user interest communities in online social networks, some researchers put forward the concept of graph database [18, 19] that has been widely used as an important tool for modelling and query graph data. Meanwhile, Subgraph matching is a basic operation in various graph operations.

However, in large-scale social networks, existing interest community evolution algorithms cannot achieve ideal results in terms of accuracy and efficiency. In this paper, a novel User Interest Community Evolution (UICE) model based on subgraph matching is proposed to accurately detect the corresponding communities in the evolution of the community. At the same time, in order to solve the problem of replacing influential users, a variant of subgraph matching is proposed, which is called Subgraph Matching with Dynamic Weight (SMDW). The model makes up for the problems of existing algorithms and achieves good results in efficiency, accuracy and scalability.

The main tasks are described as follows:

- (1) By using a topic scoring method with authority and the minimum distance of posts [6], the number of users that make up the core subgraph can be calculated to ensure these users have the greatest influence.
- (2) A novel User Interest Community Evolution (UICE) Model based on subgraph matching is proposed for accurately detecting the corresponding communities in the evolution of the community.
- (3) In order to solve the problem of replacing influential users, a variant of subgraph matching SMDW is proposed.
- (4) The experiments on real-world datasets demonstrate our model obtains better clustering accuracy, higher operating efficiency and more scalable compared with existing algorithms.

The rest of the paper is organized as follows. In section II related work on dynamic community evolution tracking and subgraph match has been discussed. In Section III and IV, our user interest community evolution model is presented based on subgraph matching in dynamic heterogeneous social networks. Our experiment results have been analyzed and discussed in Section V. The conclusions have been given and our future work has been outlined in the last section.

### **Related work**

Social networks will generate huge amounts of data every day. Hundreds of applications are deployed at the edge to consume this data using edge computing. Some necessary processing on these data is made to get better results. Community tracking and other analysis in dynamic heterogeneous social networks are important challenges that need to be solved urgently. At present, analyzing the formation and change process of communities in dynamic networks has become a research hotspot. There are two main points in the UICE model that are proposed in this paper: dynamic community evolution tracking and subgraph matching.

### Dynamic community evolution tracking

For the last few years, community research has focused on the process of community structure change in dynamic networks. In the early days, when people studied the community structure in the network, they considered the complex networks to be static. However, virtually any complex network, especially online social networks, is changing over time. Due to the dynamic nature of social networks, the process of studying community evolution is more practical [20]. By studying the dynamics of the community and its evolution process in the network, it is of great help to better understand the internal structure of the network. Thus, the research on

community evolution mainly includes the following two categories: community discovery in dynamic networks and community evolution in dynamic networks.

### Community discovery in dynamic networks

Sattari et al. [17] proposed a label propagation method based on cascaded information diffusion to detect overlapping communities in dynamic social networks named CIDLPA. This method divides nodes into two categories according to the degree of influence, which greatly reduces the possibility of creating a monster community in a dynamic network. Lin et al. [18] proposed a framework named FacetNet which combines community extraction and evolution extraction in a unified process. The framework uses the acquired network data and the prior distribution of historical community structures to estimate the community structure and uses maximum a posteriori estimates to describe the problem. However, the above methods have the following shortcomings: the number of groups needs to be determined in advance, and cannot approach the real value expressed by the lower value of normalized mutual information; the uncertainty and randomness of label propagation lead to the low accuracy and stability of the group.

### Community evolution in dynamic networks

Folino [21] proposed an evolutionary clustering algorithm based on multi-objective optimization named DYNMOGA. The clustering framework is a multiobjective genetic algorithm, which effectively balances the time overhead and historical cost, and significantly improves the quality of the cluster, does not need a balance factor to automatically discover the number of communities. Yang [22] proposed a framework for modeling the transition of community memberships for individual nodes based on Bayesian inference, named DSBM. This framework unifies the community and its evolution with the probability generation model and uses the Bayesian approach to give a reliable prediction of community memberships. Messaoudi [23] proposed a multi-objective Bat Algorithm to get high-quality solutions which generate the initial population using the mean-shift algorithm. However, the above methods execute too long due to the need to generate an initial population randomly and are not suitable for processing large amounts of social network data.

### Subgraph matching

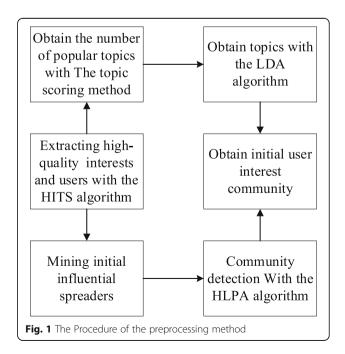
Existing subgraph matching methods could be roughly classified into two categories: exact subgraph matching and fuzzy subgraph matching. Exact subgraph matching requires that all nodes and edges match exactly. The subgraph isomorphism algorithm [24] and the VF2 [25] algorithm are classic algorithms that do not utilize any

index structure, so the cost is usually high for large graph databases. Han et al. [26] proposed a subgraph search solution TurboISO to introduce candidate region search and combination ranking strategy. However, existing exact subgraph matching does not consider the similarity of node elements. In the case of many candidate nodes, it is very expensive to find isomorphic subgraphs.

Fuzzy subgraph matching allows some nodes or edges mismatch. Closure-tree [27] is the first method supporting both exact subgraph matching and fuzzy subgraph matching. Khan et al. [28] proposed a subgraph matching technique NeMa based on neighborhood to query the real network. Li et al. [29] proposed an efficient approximate subgraph matching algorithm in fuzzy RDF graph SM-RDF, which is equivalent to a search over subgraphs of fuzzy graphs. However, none of the abovementioned methods can better model many real-world problems, because the user usually does not know all the elements of the graph database node, which makes it impossible to give the complete query conditions.

### **Preprocessing method**

The data in social networks are large and complex. In order to filter out the useful data from the complicated data and obtain the core subgraph, the data need to be preprocessed. The preprocessing method mainly includes the following steps as Fig. 1.



# Extracting popular interests and high-quality users based on HITS algorithm

Popular interests will attract more high-quality users, and the interest of high-quality users will often develop into popular interests. Popular interests get more highquality user reviews and recommendations than ordinary interests, and high-quality users' recommendations can make this interest spread more widely in social networks. In this paper, the HITS algorithm [10] is introduced, taking into account the inextricable link between users and interests, so as to extract popular interests and high-quality users. This will eliminate the less-influential users and unpopular interest. In the HITS algorithm, the authority score is represented user's significance, and the hub score is denoted interest's popularity. The users with high authority scores are called core users. The final scores can be iteratively calculated for each user and interest and can intuitively show the importance of user and the popularity of interest.

# A topic scoring method with authority and minimum distance of posts

Due to errors whilst filtering the topics and ranking popularity, the efficiency and accuracy of influence maximization lead to a poor result in existing topic detection methods. An automatic topic scoring method [10] is introduced in this paper. Because representative posts on a topic usually have higher authority, and there is usually a response distance between posts on a social network, topic scoring can be used to filter topics and score the popularity of topics. The representative posts are used for topic filtering and topic popularity ranking. So, K posts distributed in the right upper part of the graph can be automatically selected as representative posts. Furthermore, based on the assumptions of our model, the representative posts are those having higher authority values and those located dispersedly. Thus, the topic scoring method in 2-dimensional space is used to automatically filter the topics and score the popularity of topics, where one dimension is the authority value of the posts, and the other is the minimum distance of the posts as defined above. In the topic scoring method, those posts located in the right upper quadrant are shown to be the representative posts. Finally, filtering the topics and ranking popularity of topics can be completed according to the representative posts.

### The topic community detection based on LDA algorithm

In this paper, the topic community detection based on the LDA algorithm is proposed. The degree of representativeness of the posts and users in every topic is described by a variety of different weights. Each topic should consist of at least one post and one user. In the subsequent analysis, the post and user's centrality value are used to calculate the prototype weight, and the topic similarity is used to assist in the division of online social networks.

### Mining initial influential spreaders

When looking for the most influential spreaders, the choice of the initial propagation users is very important. It is related to the trend and scope of information propagation. Influence affects the user's thoughts and behaviours. When two users with different interests interact, it may cause users with low influence to change their decision. The reason why social networks have such a strong influence is that people are always inevitably affected by others, especially those who trust. The magnitude of this influence is also determined by different user characteristics, preferences, relationships, and actions. As stated previously, highly popular posts are more likely to attract the attention of high-quality users, and those users who forward or comment on such posts will make these posts attract more attention. Hence, identifying influential communicators effectively and efficiently in social networks are also a very important task. Therefore, the three-step model [10] is introduced to solve this problem: (1) extract the hub value of posters; (2) calculate the degree centrality of posters; (3) Combine the global feature and local feature to determine the initial influential spreaders. Typically, the initial influential spreaders in a social network are users with more neighbour connections. Under this premise, this paper proposes a unique influence measurement method based on user interaction hub value and degree value. To a large extent, this method can greatly improve the recognition rate of the initial influential spreaders and the final influential spreaders further enhance influence range of maximum impact.

# A user interest community detection method based on HLPA algorithm

The LPA algorithm has significant shortcomings: the result is very unstable due to the random selection scheme, and it is difficult to guarantee the quality of the community. Hence, in order to be able to select the core node set in a short period of time, the impact of the largest solution in the selection of community classification algorithm in addition to the time complexity of the low, but also should be stable and reasonable and community detection of quality assurance. This paper proposes a stable and high-quality algorithm based on node influence called HLPA [30]. To be more specific, the HLPA algorithm assigns a unique interest label to each user node, and then updates the user node's interest label in descending order of interest popularity. In each round of label update, the label is updated from the label of the adjacent node of the most influential user node. If the

labels are different, the label with the highest influence is selected for propagation by calculating the influence of the label. After all the user nodes' interest tags have been updated, a stable community will be obtained. If there are more user nodes without interest tags in the network, or the relationship between the user nodes is not enough, the network will be divided into many independent communities, making the final community detection results inaccurate. The HLPA algorithm can be used to recommend the corresponding label to the unlabeled node, and the relationship before the number of nodes is increased, so that the high quality of the community detection result can be obtained, and the data sparsity problem can also be solved.

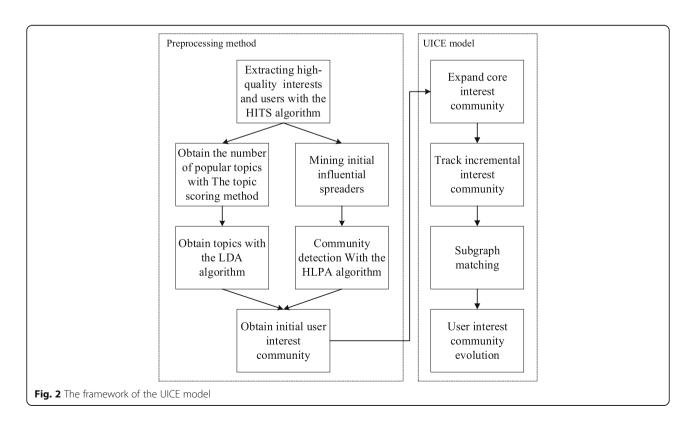
# A user interest community evolution model based on subgraph matching

Community evolution research is an important part of community structure research. Evolution is the basic characteristic of real networks. The communities in social networks are developing continuously with time, which is the result of the interaction between the network's own structure and frequent interactions occurring on it. In the research of community evolution, a community evolution model based on the historical characteristics of the community in the network is built, and the possible changes in the future are further predicted. The study of community evolution can also

facilitate researchers to analyze changes in user interest and predict user behavior and hotspot trends in the future. The information presented in social networks will be updated rapidly with the passage of time and various social events, resulting in changes in users' social relations, behaviors and interests, which may lead to changes in the community to which users belong. For example, a user in the community who follows other users in the same community or comments and forwards their posts will bring the connections between users in the community closer. Conversely, a user who frequently follows users in other communities, or comments and forwards their posts can cause the user to move to another community. In this paper, referring to the subgraph increment method proposed by Liu et al. [31], a novel User Interest Community Evolution (UICE) Model based on subgraph matching is proposed to accurately detect the corresponding communities in the evolution of the community. The model has high efficiency and good scalability. The community obtained in the preprocessing method will be used as the initial community to participate in the subsequent community evolution analysis. The steps included are as Fig. 2.

### Core interest community expanding

The core subgraph obtained by the preprocessing method is needed to be extended. For an ordinary user node u not included in core subgraph, all adjacent user



nodes of user node u are traversed. If its adjacent user nodes belong to multiple interest communities, the intimacy values of the different interest communities are compared, and the user node u is added to the interest community with the largest intimacy value. The intimacy value of the interest community is defined as:

$$V_{in} = \frac{H_{iC}^{*_{in}}}{H_{iC}^{*_{in}} + H_{iC}^{*_{out}}} \tag{1}$$

where  $H_{iC}^{*in}$  is the sum of hub value of user nodes within interest community,  $H_{iC}^{*out}$  is the sum of hub value of user nodes adjacent to user nodes within interest community. According to the principle of definition, the time cost of the extension is also very low.

### Incremental interest community evolution tracking

Since the community structure for detecting dynamic social networks differs from that for detecting static networks, an algorithm named IIC is used to update the community structure on different timestamps. When tracking and updating the community structure, the algorithm uses previously given information instead of recalculating. Because community detection is not required in every subgraph, the efficiency of the algorithm is greatly improved.

One of the key ideas of the IIC algorithm is to update the community structure by updating a subgraph  $\Delta S_{t+1}$  between consecutive moments. The scale of the core subgraph is much smaller than that of the whole graph, which guarantees the time complexity of the incremental algorithm and the good consistency of the community structure in the neighborhood time. Our framework performs very well in the large-scale data processing.

Through the preprocessing method, the core user node set  $SV_{t+1}$  and core edge set  $SE_{t+1}$  (means the relationship between the user nodes) can be obtained at each time step. The operations on the core user nodes and edges are defined as follows:

$$SV_{t+1} = SV_t - SV_{del} + SV_{new} \tag{2}$$

$$SE_{t+1} = SE_t - SE_{del} + SE_{new} \tag{3}$$

$$SV_{del} = SV_t - SV_{t+1} \tag{4}$$

$$SV_{new} = SV_{t+1} - SV_t \tag{5}$$

$$SE_{del} = SE_t - SE_{t+1} \tag{6}$$

$$SE_{new} = SE_{t+1} - SE_t \tag{7}$$

where  $SV_{del}$  denotes the deleted core user node set;  $SV_{new}$  denotes the new core user node set;  $SE_{del}$  denotes removed edges;  $SE_{new}$  is the new edges set.

Birth, growth, atrophy, merger, splitting and death are all events related to community evolution. The change

of the core subgraph from time t to t+1 is shown in Fig. 3. These community evolution events are caused by different reasons, among which the birth, growth and merger are caused by the addition of user nodes or edges, and the atrophy, splitting and death are caused by the deletion of user nodes or edges.

In view of the above, the following steps are taken to track incremental community evolution:

### - Step 1: Delete core user nodes and edges

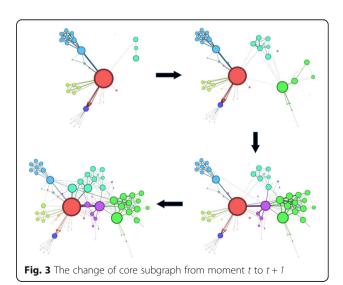
The removed user node set  $SV_{del}$  can be obtained from the formula 4, and the removed edge set  $SE_{del}$  can be obtained from the formula 6. At time step t, the user nodes have been assigned to different communities. Then these user nodes need to be deleted from the core subgraph at time step t+1. And the same work needs to be done to  $SE_{del}$ .

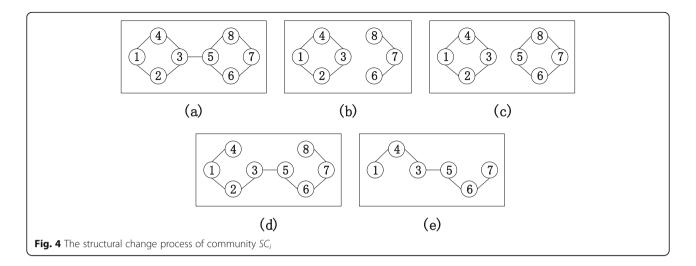
 Step 2: Process the rest user node in subgraph at time step t + 1.

For each community in the core subgraph at time step t+1, its connectivity is computed. For example, the structure of a community  $SC_i$  is shown in Fig. 4(a). Its connectivity is computed after deleting user nodes and edges in  $SC_i$ . Then if  $SC_i$  splits to two subgraphs,  $SC_i$  will be deleted and two new communities  $SC_{i+1}$  and  $SC_{i+2}$  are created as shown in Fig. 4(b) and (c). If  $SC_i$  does not split, nothing will be done as shown in Fig. 4(d) and (e).

### - Step 3: Add new user nodes

The set of new user nodes  $SV_{new}$  and the set of neighboring user nodes of each user node in  $SV_{new}$  can be obtained from formula 5. If the adjacent user nodes of a





user node u in  $SV_{new}$  belong to multiple communities, the intimacy values between u and different communities are compared. User node u joins to the community with the highest intimacy value. If the adjacent user nodes of u do not belong to any existing communities in the core subgraph at time step t+1, a new community will be created, and the user node u will be added to this community. If the user node u is only associated with one community, then u is added to the community.

### Step 4: Add new edges

The set of new edges  $SE_{new}$  can be obtained from the formula 7. The source node and the target node of an edge e in  $SE_{new}$  are found. If they belong to the same community, e is added to the community; if they do not belong to the same community, the two communities are linked. An initialized community structure is generated at this step.

### Step 5: Merge communities

For each community in the core subgraph at time step t+1, the correlation between every two communities is computed. Specifically, the correlations of two communities  $KC_i$  and  $KC_j$  have been calculated. The community correlation Ne  $(KC_i \cap KC_j)$  is defined as the number of edges between community  $KC_i$  and  $KC_j$ . Ne  $(KC_i)$  denotes the number of edges in  $KC_i$ . The two communities are merged, when Ne  $(KC_i \cap KC_j) > 0.2^*$  Ne  $(KC_i)$  and Ne  $(KC_i \cap KC_j) > 0.2^*$  Ne  $(KC_i)$ .

Step 6: Judge the validity of the core subgraph at time step t + 1.

When the core subgraph is obtained through steps 1–5, its CSM value needs to be calculated.

Community structure stability (CSM) is an indicator used to measure community effectiveness.

$$CSM_{m} = 1 - \frac{\sum_{t=1}^{m} \Delta |SE_{t,t-1}|}{|SE_{0}|}$$
 (8)

where  $|SE_0|$  denotes the number of core edges at the initial time step,  $\Delta |SE_{t,t-1}|$  denotes a change in the number of core edges between two adjacent time steps. If  $CSM \le \theta$ , core subgraph keeps unchanged; If  $CSM > \theta$ , the core subgraph is redetected by preprocessing method on the graph.  $\theta$  is a given threshold.

### Subgraph matching

In social networks, people's interests change every day. When the interest of the users in the core subgraphs changes, the influential replacement users are needed to be quickly found. The new core subgraph is retrieved by reusing the preprocessing method, but this approach will result in very low algorithm efficiency. In this paper, a variant of subgraph matching is proposed, which is called Subgraph Matching with Dynamic Weight. In the SMDW query, each graph node has a collection of elements instead of a label, and each element corresponds to a dynamic weight, the weight of the element is specified by the user at the time of the query. Specifically, given a query graph Q with n nodes  $\{u_1, u_2, ..., u_n\}$ , The SMDW query can find all subgraphs X in the graph database G that contains n nodes  $\{v_1, v_2, ..., v_n\}$ , Satisfied: (1) The dynamic weight of  $S(u_i)$  and  $S(v_i)$  is greater than the user-specified threshold, where  $u_i$  corresponds to  $v_i$ ,  $S(u_i)$  and  $S(v_i)$  denote the set of elements of  $u_i$ and  $v_i$ , respectively,  $i, j \in \{1, 2, 3, ... n\}$ . (2) X and Q are structurally isomorphic.

A network can be modelled as a graph  $G = \langle V(G), E(G) \rangle$ , called a data graph, where V(G) denotes a set of

vertices,  $E(G) \subseteq V(G) \times V(G)$  is a set of edges, and each vertex  $v \in V(G)$  has some elements, denoted as S(v), for all nodes. The complete set of elements is recorded as S(G). Similarly, the query graph can also be expressed as S(G). This article discusses the SMDW queries in undirected simple graphs, without loss of generality, our algorithm can be extended to directed simple graphs.

Definition 1. Dynamic weight is a concept in fuzzy set theory. It is an effective measure to describe uncertainty relations. It is defined as:

- (1)  $0 \le DW(y/x) \le 1$ ;
- (2) if  $x \le y$ , DW(y/x) = 1;
- (3) if  $x \le y \le z$ ,  $DW(x/z) \le DW(x/y)$ ;
- (4) if  $x \le y$ , there is an DW(x/z) < DW(y/z) for any  $z \in L$ .

Then the DW is called the dynamic weight on the partial order set  $(L_2 \le)$ .

Definition 2. Set Dynamic Weight. For sets X and Y, X is a non-empty set, and SDW(Y/X) is defined as:

$$SDW(Y/X) = \frac{|Y \cap X|}{X} \tag{9}$$

where |\*| indicates the number of elements in the collection, and  $\cap$  indicates the set intersection operation. It is easy to verify that SDW satisfies:

- (1)  $0 \le SDW(Y/X) \le 1$ ;
- (2) if  $X \subseteq Y$ , SDW(Y/X) = 1;
- (3) if  $X \subseteq Y \subseteq Z$ ,  $SDW(X/Z) \le SDW(X/Y)$ ;
- (4) if  $X \subseteq Y$ , for any non-empty set Z, there is  $SDW(X/Z) \le SDW(Y/Z)$ .

Called SDW is the set dynamic weight.

Definition 3. Dynamic Weighted Set. For sets X and Y, a is an element within X or Y, and W(a) denotes the weight of element a, which is specified by the user before each query, where  $0 \le W(a) \le 1$ . DWS(Y/X) is defined as:

$$DWS(Y/X) = \frac{\sum_{a \in Y \cap X} W(a)}{\sum_{a \in X} W(a)}$$
 (10)

where *DWS* is called Dynamic Weighted Set. For simplicity, the dynamic weight in this paper refers to Dynamic Weighted Set without special explanation.

Definition 4. Subgraph Match with Dynamic Weight (SMDW). Given the data graph G,  $V(G) = \{v_1, ..., v_m\}$ , the query graph Q,  $V(Q) = \{u_1, ..., u_n\}$ , the user-specified weight of each element and the inclusion threshold  $\tau$ ,

and Only when the following three conditions are met, it means that subgraphs X and  $V(X) = \{v_1, ..., v_n\}$  of Q and G are matched based on the subgraphs of the degree:

- (1) There is a bijective function f, for each  $u_i \in V(Q)$  and  $v_i \in V(X)$ , there is  $f(u_i) = v_i$ , where  $1 \le i$ ,  $j \le n$ ;
- (2)  $DWS(S(u_i), S(v_j)) \ge \tau$ , where  $S(u_i)$  and  $S(v_j)$  represent the set of elements of  $u_i$  and  $v_j$  respectively, and  $DWS(S(u_i), S(v_j))$  denote the dynamic weights of  $S(u_i)$  and  $S(v_i)$ ;
- (3) For any side  $(u_i, u_k) \in E(Q)$ , there is  $(f(u_i), f(u_k)) \in E(X)$ , where  $1 \le k \le n$ .

Since the SMDW has nothing to do with the edge, our method adapts to both the directed graph and the undirected graph.

In this paper, the dynamic weight is used as the measure of whether the node is matched. There are two reasons: (1) dynamic weight can better model many practical problems. For example, users often cannot know all the elements of the node feature set in the data graph. (2) Give each element different weights. The real response is that the user pays different attention to each element. In practical applications, the user specifies the query graph and the weight of each element.

### **Experiments**

### **Experiment settings and datasets**

The experiment is carried out on a computer equipped with 4.0 GHz CPU and 16G memory. Our datasets are collected from Twitter through the Twitter API. The datasets selected in this paper filter popular posts and high-impact users on Twitter from April 2015 to October 2019. The datasets contain the following contents: user information, post information and plain text review information. The specific descriptions of the datasets are shown in the following Table 1.

In order to ensure the validity of the results, each experiment randomly selects 500,000 user nodes from the dataset, and takes the average after 5 runs. Data processing is completed through the local edge computing layer to ensure the speed and safety of data processing.

Table 1 Descriptions of the datasets

Dataset statistics	Quantity
Number of reviews	25,013,682
Number of users	4,621,894
Number of posts	2,184,676
time span	Apr. 2015 - Oct. 2019

### Comparative methods

In this paper, a user interest community evolution (UICE) model based on subgraph matching is proposed. At the same time, it also solves the problem that core users need to be replaced because of the change of users' interests, which often occurs in the evolution of interest communities. From these two aspects, the performance of the method is evaluated by comparing with the existing algorithms.

First aspect: dynamic user interest community evolution tracking

- (1) FacetNet algorithm [18]: this algorithm generates association with random block model, and uses the probability model of Dirichlet distribution to analyze the evolution of association.
- (2) DSBM algorithm [22]: this algorithm is a Bayesian inference-based framework for finding communities and capturing community evolution in dynamic social networks.
- (3) CIDLPA algorithm [17]: this algorithm divides nodes into two categories according to the degree of influence to detect communities in dynamic social networks.

Second aspect: core user replacement based on subgraph matching

- (1) NeMa algorithm [28]: this algorithm is a novel graph query framework by subgraph matching, which allows ambiguity on both structure and node labels.
- (2) DYNMOGA algorithm [21]: This algorithm automatically discovers the number of communities using the genetic algorithm to select the optimal solution.
- (3) SM-RDF algorithm [29]: This algorithm is equivalent to a search over subgraphs of fuzzy graphs that have high possibility to match with a given query graph.

### **Evaluation measures**

In order to compare User Interest Community Evolution (UICE) model with other methods, the validation measure NMI (Normalized Mutual Information) is introduced [32]:

$$NMI(A, B) = \frac{-2\sum_{t=1}^{C_A} \sum_{j=1}^{C_B} C_{ij} \log(C_{ij}N/C_{i.}C_{.j})}{\sum_{t=1}^{C_A} C_{i.} \log(C_{i.}/N) + \sum_{t=1}^{C_B} C_{.j} \log(C_{.j}/N)}$$

where  $C_A$  denotes the number of communities in A, and  $C_B$  denotes the number of communities in B,  $C_i$  denotes the total number of rows in matrix C,  $C_j$  denotes the total number of columns in matrix C, and N denotes the number of nodes.

In order to compare the UICE model with other methods, five validation measures are introduced: community precision (*Precision*), community recall (*Recall*), *F-measure*, the core users' influence (*I*) and search time (*T*):

$$Precision = \frac{|x \cap y|}{|x|} \tag{12}$$

$$Recall = \frac{|x \cap y|}{|y|} \tag{13}$$

$$F\text{-}measure = 2 \frac{Precision*Recall}{Precision + Recall}$$
 (14)

$$I = \frac{U_i}{U} \tag{15}$$

where  $U_i$  denotes the number of users affected in the community, and U denotes the total number of users in the community.

### Parameter experiment

(1) The initial influential spreaders analysis

In the preprocessing method, the number of core users in the subgraph needs to be determined. To this end, a topic scoring method with permissions and minimum post distance is proposed to differentiate the importance of users on each hot topic.

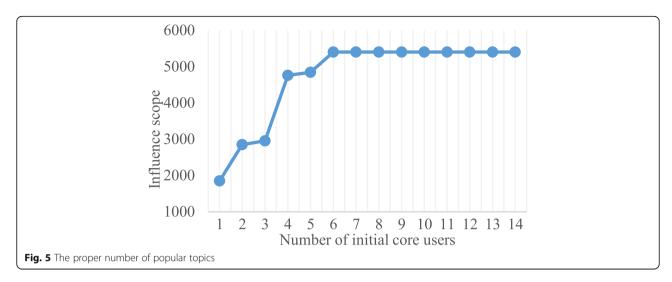
In the experiment, the number of initial core users is determined by counting the number of users affected. The more users it affects, the more influential it is.

As shown in Fig. 5, a different number of hot topics can also be found by setting a different number of initial influential users. As the number of initial core users increases, the influence scope increases. When the initial number of core users is 6, the influence scope reaches a maximum of 5400 and no longer increases. Therefore, when the initial core user number is set to 6, the maximum influence scope can be obtained.

### Result analysis

(1) Dynamic user interest community evolution tracking

In this section, our datasets are obtained from Twitter. These records constitute a small social network; there are various relationships between users, such as

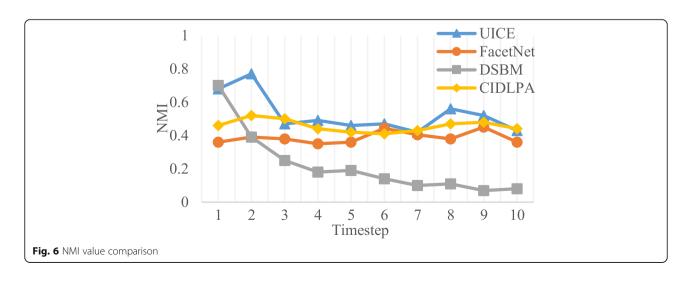


following, forwarding, replying, and so on. The UICE model is compared with other existing algorithms, and selected *NMI* which is mentioned above as the evaluation criteria. The result is shown below.

As shown in Fig. 6, the NMI value of UICE is sometimes lower than that of DSBM at the initial time step. Starting from the second step, the NMI value of UICE is significantly higher than that of FacetNet, DSBM and CIDLAP in most cases. But the NMI value of CIDLPA is occasionally higher than that of UICE. This is since the CIDLPA adopts the cascade information diffusion model on the label propagation algorithm, which improved the label propagation approach and accuracy. This largely offsets the adverse effects of the random block model. The NMI value of DSBM decreases with time, and it is almost impossible to reveal community structure after timestamp 5. In the smooth time framework, UICE achieves a higher clustering quality and reduces the deviation from the ground truth. This is since the DSBM is a probabilistic generative model based on Bayesian inference. Good results can be achieved in the initial time step, but with the passage of time, the effect is getting worse and worse. FacetNet uses random block model to generate associations, and a probabilistic model based on Dirichlet distribution to analyze the evolution of associations, which results in low accuracy. Therefore, the comparison results show that FacetNet is inadequate for the evolution tracking of user interest communities in dynamic social networks. UICE overcomes the shortcomings of the first three algorithms, uses HITS-based preprocessing method to effectively improve the local information quality, and uses improved label propagation algorithm to improve clustering effect and convergence speed.

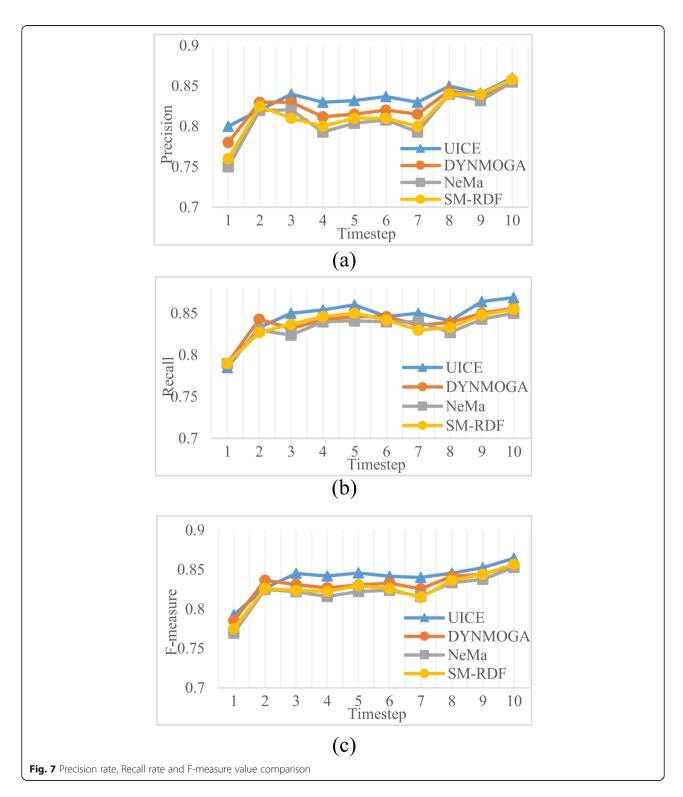
### (2) Core user replacement based on subgraph matching

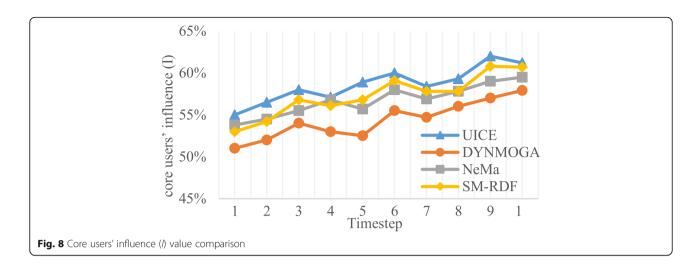
In social networks, people's interests may change at any time. It is unavoidable to replace core users based on subgraph matching when tracking dynamic user



interest community evolution. The UICE model is compared with other existing algorithms, and *Precision, Recall, F-measure, I*, which are mentioned above are selected as the evaluation criteria. The result is shown below.

As shown in Fig. 7, the results of the three methods are not much different. This is because the three methods use the respective preprocessing methods to process the data, and the core subgraphs can be obtained ideally without considering the efficiency.





However, there are many options for core user replacement, but how to make the core subgraph after replacement have the most influence is one of the important factors determining the quality of the method. The result is shown below.

As shown in Fig. 8 and Table 2, the DYNMOGA algorithm is far worse than the other algorithms in core user replacement. This is mainly due to the fact that the DYNMOGA algorithm uses a genetic algorithm and graph-based coding makes the time complexity higher. It can not handle large scale networks well. At the same time, due to the dynamic nature of the network, the randomness generated by individuals is large, which does not guarantee optimal results. The NeMa algorithm transforms the neighborhood of each node into a multi-dimensional vector, and then uses inference algorithm to identify the best graph matching, so this algorithm has good efficiency, but the accuracy still needs to be improved. The SM-RDF algorithm uses a path-based solution to improve efficiency, which decomposes the query into a set of possibly overlapping paths, finds matches of individual paths, among which a subset of possible matches with good selectivity is picked as a candidate by certain context criteria. But it is very timeconsuming to traverse the graph in the preprocessing stage, so the efficiency still needs to be improved. Although the UICE model, the NeMa algorithm and

Table 2 The comparison of time efficiency

Search time (T)
5.5 h
6 h
7 h
10 h

the SM-RDF algorithm are all based on subgraph matching methods, the UICE model uses HITS and LPA in the pre-processing which are used to ensure that the most influential replacement users can be obtained.

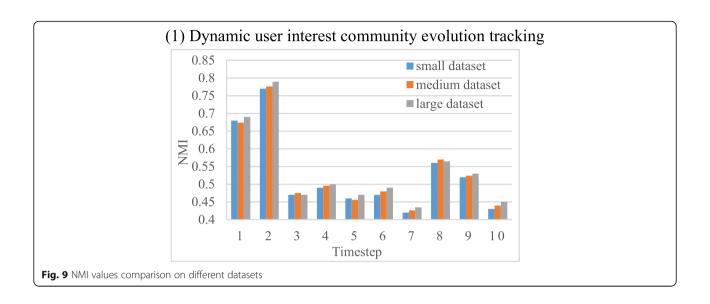
### Scalability analysis

At present, the popularity of social networks has caused its data volume to grow rapidly. To analyze the evolution of interest communities in such a huge amount of data, it is necessary for the algorithm to have strong scalability. In order to verify the scalability of the UICE model proposed in this paper, the existing datasets are divided into the large dataset, the medium dataset and the small dataset according to the number of user nodes for experiments. The previous experiments are all completed on the dataset of 500,000 user nodes, which is called the small dataset. The medium dataset is set to 1,000,000 user nodes, and the large dataset contains all the more than 2,000,000 user nodes. The result is shown below.

As shown in Figs. 9 and 10, the results of the UICE model on datasets of different sizes are not much different in aspect of dynamic user interest community evolution tracking and core user replacement based on subgraph matching. In general, the larger the dataset, the better the results. This is because when the data set is large, more relevant information can be obtained, such as follow, repost, comment, etc., so that more accurate results can be obtained. This further proves that the UICE model proposed in this paper has very good scalability.

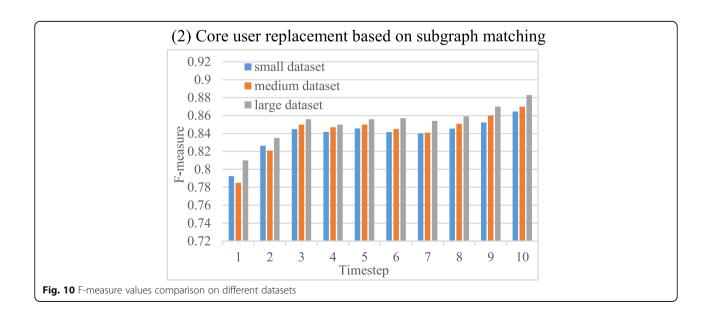
### **Conclusion**

The popularity of social networks makes it an important platform for people to share and deliver



information. Internet of People such as online social networks is the next frontier for the Internet of Things. The combination of social networking and edge computing technology has important application value and is the development trend of future networks. In order to meet the basic needs in real-time business, application intelligence, security and privacy protection, social networking and edge computing technology are combined. In this paper, a novel UICE Model based on subgraph matching in the context of edge computing is proposed to accurately detect the corresponding communities in the evolution of the community. The proposed model provides

a solution at each time step that provides the best trade-off between clustering accuracy from current data and minimum drift from one step to the adjacent time step. Besides the core user subgraph is obtained by the preprocessing method based on HITS and LPA. The model adopts the incremental subgraph method and introduces the core subgraph to infer the core community, so that it can quickly capture the community evolution events, including formation, dissolution, evolution, etc., thereby greatly reducing the running time. Finally, the experimental results demonstrate that our proposed model has achieved better performance than the state-of-art algorithms.



### Acknowledgements

The work reported in this paper has been supported by the National Natural Science Foundation of China Program (61502209 and 61502207), the Natural Science Foundation of Jiangsu Province under Grant BK20170069 and UK-Jiangsu 20-20 World Class University Initiative programme.

### Authors' contributions

Liang Jiang, Lu Liu, Jingjing Yao, and Leilei Shi developed the idea of the study, participated in its design and coordination and helped to draft the manuscript. Liang Jiang and Leilei Shi contributed to the acquisition and interpretation of data. Lu Liu provided critical review and substantially revised the manuscript. All authors read and approved the final manuscript.

### Authors' information

**Liang Jiang** received the B.S. degree from the Nanjing University of Posts and Telecommunications, China, in 2007, and the M.S. degree from Jiangsu University, Zhenjiang, China, in 2011, where he is currently pursuing the Ph.D. degree with the School of Computer Science and Telecommunication Engineering. His research interests include OSNs, computer networks, and network security.

**Lu Liu** received the M.S. degree from Brunel University and the Ph.D. degree from the University of Surrey. He is currently a Professor of Distributed Computing with the University of Leicester, U.K. His research interests are in areas of cloud computing, social computing, service-oriented computing, and peer-to-peer computing. Prof. Liu is a fellow of the British Computer Society.

Jingjing Yao received the B.E. degree from Jiangsu University, Zhenjiang, China, in 2011, and the D.M. degree from Jiangsu University, Zhenjiang, China, in 2016. Her research interests include complex network, information dissemination.

Leilei Shi received the B.S. degree from Nantong University, Nantong, China, in 2012, and the M.S. degree from Jiangsu University, Zhenjiang, China, in 2015, where he is currently pursuing the Ph.D. degree with the School of Computer Science and Telecommunication Engineering. His research interests include event detection, data mining, social computing, and cloud computing.

### Funding

This work was supported in part by the National Natural Science Foundation of China under Grant 71701082, in part by the Natural Science Foundation of Jiangsu Province under Grant BK20170069, in part by the U.K.—Jiangsu 20–20 World Class University Initiative Programme, in part by the U.K.—China Knowledge Economy Education Partnership, in part by the Postgraduate Research and Practice Innovation Program of Jiangsu Province under Grant KYCX17\_1808, and in part by Natural Science Research Projects of Jiangsu Higher Education Institutions under Grant 19KJB520027.

### Availability of data and materials

The datasets used or analysed during the current study are available from the corresponding author on reasonable request.

### Competing interests

The authors declare no conflict of interest.

### **Author details**

<sup>1</sup>School of Computer Science and Telecommunication Engineering, Jiangsu University, Zhenjiang, China. <sup>2</sup>Jiangsu Key Laboratory of Security Technology for Industrial Cyberspace, Jiangsu University, Zhenjiang, China. <sup>3</sup>School of Informatics, University of Leicester, Leicester, UK. <sup>4</sup>School of Economy and Finance, Jiangsu University, Zhenjiang, China.

## Received: 27 July 2020 Accepted: 29 November 2020 Published online: 14 December 2020

### References

 Ren J, Zhang D, He S, Zhang Y, Li T (2019) A survey on end-edge-cloud orchestrated network computing paradigms: transparent computing, mobile edge computing, fog computing, and cloudlet. ACM Comput Surv 52(6):125

- Khan WZ, Ahmed E, Hakak S, Yaqoob I, Ahmed A (2019) Edge computing: a survey. Futur Gener Comput Syst 97:219–235
- Gai K, Xu K, Lu Z, Qiu M, Zhu L (2019) Fusion of cognitive wireless networks and edge computing. IEEE Wirel Commun 26(3):69–75
- Zhang Z, Cui P, Zhu W (2020) Deep learning on graphs: a survey. IEEE Trans Knowl Data Eng in press
- Jiang L, Shi L, Liu L, Yao J, Yuan B, Zheng Y (2019) An efficient evolutionary user interest community discovery model in dynamic social networks for internet of people. IEEE Internet Things J 6(6):9226–9236
- Li Z, Chen R, Liu L, Min G (2016) Dynamic resource discovery based on preference and movement pattern similarity for large-scale social internet of things. IEEE Internet Things J 3(4):581–589
- Shi L, Liu L, Wu Y, Jiang L, Hardy J (2017) Event detection and user interest discovering in social media data streams. IEEE Access 5:20953– 20964
- Liu L, Antonopoulos N, Minghui Z, Zhan Y, Ding Z (2016) A Socioecological model for advanced service discovery in machine-to-machine communication networks. ACM Trans Embed Comput Syst 15:1–26
- Hu B, Wang H, Yu X, Yuan W, He T (2019) Sparse network embedding for community detection and sign prediction in signed social networks. J Ambient Intell Humaniz Comput 10(1):175–186
- Shi L et al (2019) Human-centric cyber social computing model for hot-event detection and propagation. IEEE Trans Comput Soc Syst 6(5):1042–1050
- Zhan XX, Liu C, Zhou G, Zhang Z (2018) Coupling dynamics of epidemic spreading and information diffusion on complex networks. Appl Math Comput 332:437–448
- Guo Y, Liu L, Wu Y, Hardy J (2018) Interest-aware content discovery in peerto-peer social networks. ACM Trans Internet Technol 18(3):39
- 13. Zhang C, Patras P, Haddadi H (2019) Deep learning in mobile and wireless networking: a survey. IEEE Commun Surv Tutorials 21(3):2224–2287
- Zhao Z, Li C, Zhang X, Chiclana F, Herrera-Viedma E (2019) An incremental method to detect communities in dynamic evolving social networks. Knowl-Based Syst 163:404–415
- Žalik KR (2019) Evolution algorithm for community detection in social networks using node centrality. In: Intelligent methods and big data in industrial applications. Springer, Berlin, pp 73–87
- Visheratin AA, Trofimenko TB, Mukhina KD, Nasonov D, Boukhanovsky AV (2017) A multi-layer model for diffusion of urgent information in mobile networks. J Comput Sci 20:129–142
- Sattari M, Zamanifar K (2018) A cascade information diffusion based label propagation algorithm for community detection in dynamic social networks. J Comput Sci 25:122–133
- Lin Y-R, Chi Y, Zhu S, Sundaram H, Tseng BL (2009) Analyzing communities and their evolutions in dynamic social networks. ACM Trans Knowl Discov Data 3(2):8
- Kong X, Shi Y, Yu S, Liu J, Xia F (2019) Academic social networks: modeling, analysis, mining and applications. J Netw Comput Appl 132:86–103
- Zhang Z-K, Liu C, Zhan X-X, Lu X, Zhang C-X, Zhang Y-C (2016) Dynamics of information diffusion and its applications on complex networks. Phys Rep 651:1–34
- Folino F, Pizzuti C (2014) An evolutionary multiobjective approach for community discovery in dynamic networks. IEEE Trans Knowl Data Eng 26(8):1838–1852
- Yang T, Chi Y, Zhu S, Gong Y, Jin R (2011) Detecting communities and their evolutions in dynamic social networks—a Bayesian approach. Mach Learn 82(2):157–189
- 23. Messaoudi I, Kamel N (2019) A multi-objective bat algorithm for community detection on dynamic social networks. Appl Intell 49(6):2119–2136
- 24. Ullmann J (1976) An algorithm for subgraph isomorphism. J ACM 23:31–42
- Cordella LP, Foggia P, Sansone C, Vento M (2004) A (sub) graph isomorphism algorithm for matching large graphs. IEEE Trans Pattern Anal Mach Intell 26(10):1367–1372
- Han W-S, Lee J, Lee J-H (2013) TurbolSO: towards ultrafast and robust subgraph isomorphism search in large graph databases. In: Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data ACM, pp 337–348
- Huahai H, Singh AK (2006) Closure-tree: an index structure for graph queries. 22nd International Conference on Data Engineering (ICDE'06), Atlanta. p 38
- 28. Khan A, Wu Y, Aggarwal CC, Yan X (2013) NeMa: fast graph search with label similarity. Proc VLDB Endow 6(3):181–192

- 29. Li G, Yan L, Ma Z (2019) An approach for approximate subgraph matching in fuzzy RDF graph. Fuzzy Sets Syst 376:106–126
- Jiang L, Shi L, Liu L, Yao J, Yousuf MA (2019) User interest community detection on social media using collaborative filtering. Wirel Netw 25(7): 4443
- 31. Liu Y, Gao H, Kang X, Liu Q, Wang R, Qin Z (2015) Fast community discovery and its evolution tracking in time-evolving social networks. In: 2015 IEEE International Conference on Data Mining Workshop (ICDMW), pp 13–20
- 32. Lancichinetti A, Fortunato S (2009) Community detection algorithms: a comparative analysis. Phys Rev E 80(5):056117

### **Publisher's Note**

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

# Submit your manuscript to a SpringerOpen journal and benefit from:

- ► Convenient online submission
- ► Rigorous peer review
- ▶ Open access: articles freely available online
- ► High visibility within the field
- ► Retaining the copyright to your article

Submit your next manuscript at ▶ springeropen.com