Community Search in Spatial Uncertain Network

Wenqian Zhang $^{1,a}$, Yingli Zhong $^{1,b}$, Yan Yang $^{1,2,c,*}$

$^1$ School of Computer Science Technology, Heilongjiang University, Harbin 150080, China
$^2$ Key Laboratory of Database and Parallel Computing of Heilongjiang Province, Harbin 150080, China

$^a$ zhangwenqian215@163.com, $^b$ zhongyingli@hlju.edu.cn
$^*,^c$ Corresponding author e-mail: yangyan@hlju.edu.cn

Abstract. Community search is to explore valuable target community structure from a large social network. In real community, every point has a geographic location information, and many edges are uncertain. Such a network is called spatial uncertain network. In order to meet the existing needs, this paper studies the community search in the spatial uncertain network so that the searched communities can be relatively close in the geographical location and the relationship between each other is also very close. Based on the spatial uncertain graph, this paper proposes K-R-$\tau$ community, and proposes a linear algorithm and a two-dimensional algorithm for community search in the spatial uncertain network. A large number of experiments are carried out on several real datasets, and the results show that the algorithm proposed in this paper is effective and accurate.

Keywords: Uncertain network, community search, geographic information, social network, spatial uncertain network

1. Introduction

 Communities generally exist in social networks, knowledge graphs and biological networks. In recent years, the topic of community search has been widely concerned. Given a query node, the goal of community search is to find the dense subgraph containing the node. With the emergence of twitter, foursquare and other geographic social networks, the topic of geographic social networks has been widely concerned [1,2,3,4]. In these networks, users are usually associated with location information (for example, their home location and check-in information). These networks are collectively referred to as spatial graphs. However, the existing community search solutions do not consider the spatial scope of the community. Many networks in the real world are uncertain, in which each edge is associated with a probability, indicating the possibility of the existence of the edge. In order to meet the existing needs, the community can be located in a relatively close geographical position and the relationship between each other is also very close, this paper proposes and studies the community search problem on the spatial uncertain graph.

In summary, the main contributions of our work are the following.
A community structure of spatial uncertain graph and K-R-τ community structure are proposed for the first time.
For efficient K-R-τ community search in spatial uncertain network, we propose a linear solution and a two-dimensional solution respectively.
A large number of experiments are carried out on several real data sets, and the results show that the algorithm proposed in this paper is effective and accurate.

2. Related Work
In the real world, networks, such as social networks, biological networks and communication networks, are usually composed of cohesive subgraphs. Mining cohesive subgraphs from network is a basic problem in network analysis, which has attracted wide attention of database and data mining. It is a basic problem in network science to find community from the network, and has been widely studied in recent decades. The classic solution [5,6] uses link-based analysis to obtain these communities. However, they don't consider location information. Some of the most recent work [7,8] focuses on identifying communities from spatial constraints, with vertices related to spatial coordinates. For example, geo community [9] is like a community, which is a graph of tightly connected vertices loosely connected to other vertices, but it is more compact in space.

In recent years, there is another related but different problem in community detection, called community search. The goal of community search is to "get communities online" based on query requests. For example, given a vertex, some existing work proposes effective algorithms to obtain the most likely community. The minimum measure is usually used to measure the structural cohesion of the community [10,11]. In [10], sozio et al. Proposed the first global lookup inclusion algorithm. In [11], a more effective local extension algorithm is proposed, which improves the query performance by using local extension technology. In addition, some recent work [12,13] also uses the minimum measure to search communities from the attribute graph. Other well-known structural cohesion measures, including k-clique, k-truss and connectivity, are also considered for online community search. But these works assuming non spatial networks and ignore the location of vertices. Therefore, it is necessary to design the algorithm of searching community from spatial network.

The mining of uncertain network has attracted a wide range of attention in data-base and data mining. Jin et al. [14] studies the mining of highly reliable connected subgraphs in uncertain network. Mehmood et al. [14] studied the cascade problem of the influence modeling the influence network as an uncertain network. Huang et al. [15] proposed the k-truss search problem on the uncertain network. Gao et al. [16] proposed a method to find RkNN on uncertain network. Li et al. [17] studies the enumeration of the largest signed group in a signed network, where the edges in the graph can be positive or negative. Li et al. [18] studied the improved algorithm of maximum group search in uncertain networks. Yang et al. [19] studies the k-core optimization algorithm of large-scale uncertain networks based on index. But their research is limited to the uncertainty network, but not in the uncertain spatial network.

In conclusion, although there are extensive researches in the fields of spatial network and uncertain network, it is necessary to study community search in spatial uncertain network.

3. Related Concepts.

3.1. Geographic Information Social Network
In recent years, with the rapid popularity of mobile devices and the rapid development of wireless networks, social networks based on geographic information have begun to pour into people's vision. Wireless networks and mobile devices can provide users with the ability to share their geographical location, so that the social network based on geographic information is booming. Geographic information social network is a new concept that adds geographic information to the traditional social network.
3.2. Uncertain Network
When our social networks are represented by graphs, the only things that can be represented are points and edges. The traditional graph is usually represented by the same relationship when it represents the relationship between edges, which can also be called unweighted graph. Such a graph can play a certain role in relatively simple social relations, but with the progress and development of our society, simple social relations are becoming less and less, and most of the complex relations are produced. Faced with this problem, the uncertain network is born. The advantage of uncertain graph is that it further distinguishes the relationship between points, that is, it gives weight to the edge between points. Compared with the unweighted graph, this graph has more information, and this information can give more practical significance, so that it can better represent the real society. Such graphs are called weighted graphs. Through the study of uncertain network, it greatly enriches the structure information of community membership.

4. Community Search Problem in Spatial Uncertain Graph
This section gives the symbols used in this paper and the meanings they represent.

(1) \( G = (V, E) \): represents spatial uncertain graph, \( V \) is the set of vertices in graph \( G \), \( E \) is the set of edges in graph \( G \).

(2) \( E (u, G) \): represents an edge in spatial uncertain graph \( G \), and one of the nodes of the edge is \( u \).

(3) \( N_u \): represents the set of neighbors of \( u \).

(4) \( D_u \): represents the number of neighbors of node \( u \).

(5) \( dist (u, v) \): represents the distance between node \( u \) and node \( v \).

(6) \( W (u, v) \): represents the weight of the edge between node \( u \) and node \( v \).

(7) \( Un (u, v) = dist (u, v) \ast (1 - W (u, v)) \).

(8) \( v_x, v_y \): represents the value of node \( v \) on the x-axis; the value of node \( v \) on the y-axis.

Definition 4-1: (K-R-\( \tau \) community): Given a spatial uncertain graph \( G (V, E) \) and a query \( q \), the induced subgraph \( C (V_C, E_C) \) satisfying the following conditions is called K-R-\( \tau \) community:

(1) Search result limit: \( |V_C| \leq K \);

(2) Distance limit: \( \forall u \in V_C, \ dist (u, q) \leq R \);

(3) Weight limit: \( \forall u \in V_C, \forall v \in V_C, \tau \leq W (u, v) \).

4.1. Linear Community Search in Spatial Uncertain Graphs
Definition 4-2: (linear community search problem): Given a spatial uncertain graph \( G (V, E) \) and parameters \( q, K, R, \tau \). The goal of the linear community search problem is to find the subgraph \( C \) in a linear way in the spatial uncertain graph \( G \), and the subgraph \( C \) satisfies the K-R-\( \tau \) community.

For the solution process of the linear community search problem on the spatial uncertain graph, the method used is to start the search from the query node, and add his neighbor nodes to the candidate set. The nodes in the candidate set are searched according to the size of \( Un \). When a new node is added to the result set, its neighbor node is added to the candidate set, and the \( Un \) value of the node and his neighbor node is calculated until the number in the result set reaches \( K \) or the candidate set is empty.

Algorithm 1 describes this process. The input of the algorithm is a spatial uncertain graph \( G \), the parameters \( q, K, R, \tau \), and the start to search for the node \( q \); the output of the algorithm is a subgraph \( C \) of no more than \( K \) nodes. First, we process the data and compare it with the query node \( q \) the nodes and distances that do not form a connected subgraph or the query node point \( q \) are greater than \( R \), or the neighbor nodes whose weight is less than \( \tau \) are deleted (1 line). Then add the neighbor nodes of the query node \( q \) to the candidate set (3 lines). Then calculate the \( Un \) value of the points in the candidate set in turn (4-6 lines). The node with the smallest \( Un \) value in the candidate set is selected and added to the result set, and his neighbor nodes are added to the candidate set (7-9 lines). Finally, the result set is returned (10 lines).

The complexity of Algorithm 1 is \( O (knm) \). \( n \) and \( m \) correspond to the number of candidate nodes and the number of neighbors in each iteration.

Algorithm 1: LCS \((G, q, K, R, \tau)\)
Input: $G$: spatial uncertain graph, $K$, $R$, $\tau$: parameter query node $q$
Output: $C$: subgraph with no more than $K$ nodes

1. The neighbor nodes whose distance from query node $q$ is greater than $R$ or whose weight is less than $\tau$ are deleted;
2. $C = \infty$, $B = \infty$: candidate set, $i = 0$
3. $B = N_q$
4. While $i < K$ && $B \neq \emptyset$
   a. For each $u \in B$
   b. Compute $Un(u, N_u)$
   c. select $v \{ v | v, u \in B \& \& \exists N_v, Un(v, N_v) \leq Un(u, N_u) \}$
5. $C = C \cup v$
6. $B = B \cup N_v$, $i++$
7. return $C$

Figure 1 The left side is the spatial uncertain graph $G$, and the right side is the corresponding location information of nodes

Example 1: As shown in Figure 1, according to algorithm 1, let the query node be node 5, $K = 4$, $R = 5$, and $\tau = 0.5$, we can get that the K-R-\(\tau\) community of node 5 is $\{5, 6, 7, 8\}$, because when we query its K-R-\(\tau\) community for node 5, we first select the nodes 10, 11 and 12 that do not meet the conditions, Exclusion, because these points cannot form a complete connected graph with node 5. Among the neighbor nodes of node 5, only the distance between node 1 and node 5 is 5.76 greater than $R$ ($R = 5$), so node 1 is excluded and the search is started. First, the remaining neighbor nodes 2, 6, 8 and 9 of node 5 are added to the candidate set. According to the calculation, $Un(5, 2) = 1.25$, $Un(5, 6) = 0.44$, $Un(5, 8) = 0.82$, $Un(5, 9) = 1.47$. The node that can get the minimum $Un$ is node 6, so it is added to the result set, and then the neighbor nodes 2, 3, 4, 7 and 8 that meets the condition of node 6 is added to the candidate set, and the node that can get the minimum $Un$ is node 7, which loops in turn, and finally the K-R-\(\tau\) community of node 5 is $\{5, 6, 7, 8\}$.

4.2. Two-Dimensional Community Search in Spatial Uncertain Graph

The study on the K-R-\(\tau\) community search problem for spatial uncertain graphs uses a linear method in 4.1, the community search is performed according to the value of $Un$ of the node. But for two different parameters such as the spatial position information and the uncertainty of the edge, we hope that they can have their own representative meaning without being mixed together. Therefore, in this section, we will use a two-dimensional approach. And we gave the following problem definition for this problem:

Definition 4-3: (two-dimensional community search problem): Given spatial uncertain graph $G$, parameters $q$, K, R, $\tau$. The goal of the two-dimensional community search problem is to find the subgraph $C$ in a two-dimensional way in the spatial uncertain graph $G$, and the nodes in the subgraph $C$ should satisfy the K-R-\(\tau\) community.
Algorithm 2: 2dCS ($G$, $q$, $K$, $R$, $\tau$)

**Input:** $G$: spatial uncertain graph, $K$, $R$, $\tau$: parameter query node $q$

**Output:** $C$: subgraph with no more than $K$ nodes

1. The neighbor nodes whose distance from query node $q$ is greater than $R$ or whose weight is less than $\tau$ are deleted

2. $C = \infty$, $B = \infty$: candidate set $i = 0$, $X = 0$, $Y = 0$;

3. $C = \{v \mid \exists v \in B, \forall u \in B, v_x \leq u_x\}$

4. $X = v_x$, $Y = v_y$, $B = N_q U N_v$;

5. While $i < K-1$ && $B \neq \emptyset$

6. select $v = \{v \mid \exists v \in B, \forall u \in B, v_x \leq u_x\}$

7. if ($v_x > X$)

8. select $v = \{v \mid \exists v \in B, \forall u \in B, v_y \leq u_y\}$

9. $X = v_x$, $Y = v_y$;

10. $B = B \cup N_v$, $C = C \cup v$,

11. return $C$

The input of algorithm 4-2 is a spatial uncertain graph $G$, the parameters $K$, $R$ and $\tau$, and the search for node $q$; the output of the algorithm is a subgraph $C$ of no more than $K$ nodes, and the distance between the nodes is used as the coordinates on the x-axis, the 1 minus weight is the y-axis. First, we process the data, and delete the neighbor nodes whose distance from the query node $q$ that does not form a connected subgraph and the query node $q$ is greater than $R$, or whose weight is less than $\tau$ (1 line). And then delete the neighbor nodes of the query node $q$ that the node with the smallest x value in the node is added to the result set. And the other neighbor nodes of the $q$ node and the neighbor nodes of the $v$ node are stored in the candidate set $B$ (2-4 lines). And then start the loop until you know that there are $K$ nodes in the result set or there are no candidate nodes in the candidate set (5 lines). And the loop content is to continue to select the node with the smallest x value in the candidate set. If select the x value of this node $v$ is greater than the previous one, then reselect the node with the smallest y value in the candidate set $B$ as the $v$ node (6-8 lines). After selecting the $v$ node, we will use our temporary variable $X$ And $Y$, respectively assign the x and y values of the $v$ node (9 lines). Updating our candidate set $B$, that is, add the neighbor nodes of $v$ to the candidate set, and also add the $v$ node just selected to the result Concentrate (10 lines). Finally, return our result set (11 lines).

![Figure 2](image-url)
Example 4-2: As shown in Figure 2, according to Algorithm 2, let the query node be node 5, K=4, R=5, τ=0.5, the K-R-τ community of node 5 can be obtained is \{5, 6, 7, 8\}, because when we query his K-R-τ community for node 5, we first exclude nodes 10, 11, and 12 that do not meet the conditions, because these points cannot form a complete connected graph with node 5. In Among the neighbor nodes of node 5, only the distance between node 1 and node 5 is 5.76 greater than R (R=5), so node 1 is excluded, and the search is started. First, the neighbor node 2, 6, 8 and 9 of node 5 is added to the candidate set. As shown in Figure 3, the x-axis represents the distance from node 5, and the y-axis represents 1-weight (in this case, the closer the point is to the origin, the more it meets our requirements). According to Figure 3, we can get the most suitable condition for node 6, add node 6 to the result set, and then add the neighbor nodes 2, 3, 4, 7 and 8 of node 6 that meet the conditions to the candidate set. The smallest one is node 7, and the loop is in turn, and finally the K-R-τ community of node 5 is \{5, 6, 7, 8\}.

This algorithm makes good use of the position information and the uncertainty of the edges in the spatial uncertain graph. In the calculation, we can divide the algorithm into three types. The first one gives priority to the position information, that is, the x-axis of the coordinates. When choosing, we can give priority to choosing a small value of x. If the value of x is smaller than the last time, then we can continue to choose according to the value of x. Similarly, the second is to give priority to the uncertainty of the edge. Considering the information of the y axis, when the y value selected each time is less than the y value selected last time, we can continue to proceed, and the third type is to select the x value and the y value in turn. This choice is both Considering the importance of location information, it also gives equal importance to uncertainty. In practical applications, we can give these two kinds of information different status to meet the needs of different users according to the needs of users.

The time complexity of the algorithm is \(O(knm)\), k is the number of searched nodes, n and m correspond to the number of candidate nodes and the calculation process of the neighbors of candidate nodes in each cycle.

5. Experiments and Results
Three real datasets are used to test the algorithm proposed in this paper. These three datasets are Gowalla, Flickr and foursquare. These datasets contain geo-graphic location information, and the weight of their edges is calculated according to the number of common neighbors. The calculation formula is \(W(u,v) = 1 - \exp(-D_{uv} / 2)\). \(D_{uv}\) represents the number of the same nodes owned by vertex u and vertex v. Gowalla datasets are all from Stanford web datasets, which can be downloaded from https://snap.stanford.edu.
The Flickr dataset is downloaded from the Flickr website [https://www.flickr.com](https://www.flickr.com). Foursquare is from [https://archive.org/details](https://archive.org/details).

Table 1 shows the specific information of Gowalla, Flickr and foursquare. The information in the table includes the number of vertices, the number of edges, and the average number of degrees in the graph.

| Datasets   | |V| | |E| | Avg(d) |
|------------|------------------|-----------------|-----------------|
| Gowalla    | 107,092           | 456,830          | 8.53            |
| Flickr     | 214,698           | 2,096,306        | 19.5            |
| Foursquare | 2,127,093         | 8,640,3512       | 8.12            |

Figure 4 compares the efficiency of LCS algorithm and 2dCS algorithm under different K values in Gowalla data set. Let R = 15, \( \tau = 0.5 \).

![Figure 4. LCS algorithm and 2dCS algorithm under different K values in Gowalla data set](image)

According to Figure 4, it can be seen that with the increase of K, the running time of the LCS and 2dCS algorithms is increasing because there are more points to search. And the LCS algorithm runs faster than 2dCS, because the LCS algorithm considers fewer conditions.

Figure 5 compares the efficiency of LCS algorithm and 2dCS algorithm under different R values in Flickr data set. Let K = 20, \( \tau = 0.5 \).

![Figure 5. LCS algorithm and 2dCS algorithm under different R values in Flickr data set](image)

According to Figure 5, it can be seen that as R increases, the running time of the LCS and 2dCS algorithms is increasing because there are more points that meet the conditions. And the LCS algorithm runs faster than 2dCS, because the LCS algorithm considers fewer conditions.

Figure 6 compares the efficiency of LCS algorithm and 2dCS algorithm under different \( \tau \) values in Foursquare data set. Let K = 20, R = 15.

![Figure 6. LCS algorithm and 2dCS algorithm under different \( \tau \) values in Foursquare data set](image)
Figure 6. LCS algorithm and 2dCS algorithm under different R values in Four-square data set

According to Figure 6, it can be seen that as $\tau$ increases, the running time of the LCS and 2dCS algorithms is decreasing, because there are fewer points that meet the conditions. And the LCS algorithm runs faster than 2dCS, because the LCS algorithm considers fewer conditions.

6. Concluding
Firstly, this paper studies the community search problem of spatial uncertain graph, and proposes a K-R-$\tau$ community structure, which not only meets the requirements of geographic information, but also meets the requirements of uncertainty. In this paper, the linear community search problem on spatial uncertain graph is studied for the first time. Its purpose is to carry out K-R-$\tau$ community search on spatial uncertain graph, and LCS algorithm is proposed to solve this problem. In this paper, the community search problem on spatial uncertain graph is studied in two-dimensional coordinates, because the former algorithm uses linear method, in other words That is to say, the spatial location information and uncertainty are changed into one data to study. In this way, it is difficult to reflect the different roles of the two information. Therefore, we put the two information into the two-dimensional coordinates to study. In this way, the importance of different information can be reflected. To solve this problem, we propose the 2dCS algorithm. And the proposed algorithm is verified on several real experimental data, the experimental results also show that the efficiency and effectiveness of the proposed algorithm.

Acknowledgments
The Natural Science Foundation of Heilongjiang Province under Grant Nos.F2018028.

References
[1] M. Sarwat et al. Lars*: an efficient and scalable location-aware recommender system. IEEE Trans. Knowl. Data Eng, 2014, 26(6):1384–1399.
[2] Y. Fang et al. Scalable algorithms for nearest-neighbor joins on big trajectory data[J]. IEEE Trans. Knowl. Data Eng, 2016, 28(3):785–800.
[3] R. Li, L. Qin, F. Ye, J. X. Yu, X. Xiao, N. Xiao, and Z. Zheng. Skyline community search in multi-valued networks[C]. SIGMOD Conference, 2018, 457-472.
[4] Wang K, Cao X, Lin X, et al. Efficient Computing of Radius-Bounded k-Cores[C]. IEEE International Conference on Data Engineering (ICDE), IEEE, 2018, 233-244.
[5] M. Newman et al. Finding and evaluating community structure in networks. Physical review E, 2004, 69(2):026–113.
[6] S. Fortunato. Community detection in graphs[J]. Physics Report, 2009, 486(3):75–174.
[7] P. Expert et al. Uncovering space-independent communities in spatial networks. PNAS. 2011, 108(19):7663–7668.
[8] Y. Chen et al. Finding community structure in spatially constrained complex network[M]. Taylor & Francis, Inc, 2015, 29(6):889–911.
[9] A. Lancichinetti et al. Limits of modularity maximization in community detection[J]. Phys Rev E Stat Nonlin Soft Matter Phys, 2011, 84(6):066122.
[10] M. Sozio and A. Gionis. The community-search problem and how to plan a successful cocktail
party. KDD, 2010, 939–948.

[11] W. Cui et al. Local search of communities in large graphs. SIGMOD Conference, 2014, 991-1002.

[12] Y. Fang et al. Effective community search for large attributed graphs[J]. Proceedings of the VLDB Endowment, 2016, 1233–1244.

[13] R. Jin, L. Liu, and C. C. Aggarwal. Discovering highly reliable subgraphs in uncertain graphs. KDD, 2011, 992-1000.

[14] Y. Mehmood, F. Bonchi, and D. Garcí’a-Soriano. Spheres of influence for more effective viral marketing. SIGMOD Conference, 2016, 711-726.

[15] X. Huang, W. Lu, and L. V. S. Lakshmanan. Truss decomposition of probabilistic graphs: Semantics and algorithms[C]. SIGMOD Conference, 2016, 77-90.

[16] Gao Y, Miao X, Chen G, et al. On efficiently finding reverse k-nearest neighbors over uncertain graphs[J]. The VLDB journal, 2017, 26(4):467-492.

[17] Li R H, Dai Q, Qin L, et al. Efficient Signed Clique Search in Signed Networks[C]. IEEE International Conference on Data Engineering (ICDE). IEEE, 2018, 245-256.

[18] R. Li, Q. Dai, G. Wang, Z. Ming, L. Qin and J. X. Yu. Improved Algorithms for Maximal Clique Search in Uncertain Networks. ICDE, 2019, 1178-1189.

[19] Yang B, Wen D, Qin L, et al. Index-Based Optimal Algorithm for Computing K-Cores in Large Uncertain Graphs[C]. IEEE International Conference on Data Engineering (ICDE). IEEE, 2019, 64-75.