Advances in Computational Intelligence (2022) 2:31
https://doi.org/10.1007/s43674-022-00041-z

REVIEW ARTICLE

Attributed community search based on seed replacement and joint random walk

Ju Li[1] · Huifang Ma[1]

Received: 12 February 2022 / Revised: 6 August 2022 / Accepted: 14 August 2022 / Published online: 1 September 2022

Abstract

Community search enables personalized community discovery and has wide applications in real-life scenarios. Existing attributed community search algorithms use personalized information provided by attributes to locate desired community. Though achieved promising results, existing works suffer from two major limitations: (i) the precision of the algorithm decreases significantly when the seed comes from the boundary regions of the community. (ii) Most attributed community search methods mainly take the attribute information as edge weights to reveal semantic strength (e.g., attribute similarity, attribute distance, etc.), but largely ignore that attribute may serve as heterogeneous vertex. To make up for these deficiencies, in this paper, we propose a novel two-stage attributed community search method with seed replacement and joint random walk (SRRW). Specifically, in the seed replacement stage, we replace the initial query node with a core node; in the random walk stage, attributes are taken as heterogeneous nodes and the augmented graph is modeled based on the affiliation of the attributes via an overlapping clustering algorithm. And finally, a joint random walk is performed on the augmented graph to explore the desired local community. We conduct extensive experiments on both synthetic and real-world benchmarks, demonstrating its effectiveness for attributed community search.

Keywords Seed replacement · Random walk · Community search · Conductance value · Attributed graph

1 Introduction

Network analysis has many applications in the field of biotechnology, physical, computer science, and social science (Li et al. 2021; Huang et al. 2017; Luo et al. 2020a). In these areas, researchers are willing to store information utilizing graphs. A graph is often defined as a data structure consisting of nodes and edges, where nodes represent entities and edges denote relationships between entities (Fang et al. 2020). Subgraph structure is one of the most important features of complex networks, and community detection is an effective way to study this feature. However, with the rapid growth of the network scale, it is difficult for community detection to explore the entire network structure in a limited time. Therefore, online rapid local community detection has recently attracted attention. This kind of research is also known as community search, which usually explores the local graph structure based on a set of query nodes given by the user.

Community search has been extensively studied since it was first introduced by Sozio and Gionis (2010). The works of community search on simple graphs focus on devising different models, such as core-based model (Fang et al. 2016; Cheng et al. 2011), truss-based model (Akbas and Zhao 2017; Wang and Cheng 2012; Huang et al. 2014), clique-based model (Yuan et al. 2017; Cheng et al. 2011a), et al. Due to the increasing complexity of real-world networks, simple graphs are not able to adequately accommodate this rich personalized information. In recent years, researchers have proposed many attributed community search methods. For attributed graphs, the entities modeled by the network nodes often have attributes that are important for understanding communities (Zhao et al. 2021). For instance, on Facebook, users can specify hobbies, location, and other information in their profiles. By combining the community models and attribute information, community search can discover semantically similar and closely linked communities. For example, Fang et al.

✉ Huifang Ma
mahuifang@yeah.net

Ju Li
2019221843@nwnu.edu.cn

[1] College of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, Gansu, China

(2017) and Fang et al. (2016) proposed the attributed community query (ACQ) problem, which is capable of detecting densely connected subgraphs that maximize the set of shared attributes. However, maximizing shared attributes set is so rigid that makes some nodes which are critical to improving the tightness of community structure are not included. Random-walk-based methods are particularly well-suited for alleviating such problems (Andersen et al. 2006; Liu and Xia 2020), but the random walk is usually utilized for community search on simple graphs. To enable walkers to explore the attributes directly, one common approach is to use the attribute information as edge weights to indicate semantic strength (e.g., attribute similarity, attribute distance, etc.). Based on this promising insight, Hsu et al. (2017) propose an unsupervised learning framework AttriRank to improve the quality of node importance ranking.

Although random walks on attributed networks have been investigated, most existing community search methods only perform well when the query nodes are from the community core region. The above problem is known as the seed-dependent problem (Chang et al. 2022). Seed-dependent demonstrates that when a query node is from the target community, the detected community will lose some nodes in the target community or include some nodes outside the target community. To motivate this work, we first sample a case dataset from DBLP network that we will consider in this paper as suggested in Fig. 1. DBLP consists citation relation, in which the node represents the scholar with research topics as attributes, and an edge between two scholars indicates that they have a citation relationship. The statistical information of the case network is shown in the table in Fig. 1. We specified the yellow node 493542 as the query node in the upper of the left community. The node-set circled via the green curve is the detected community using the well-known PageRank-Nibble (PRN) (Tong et al. 2006) algorithm. It is obvious that the detected result is not the real target upper left community. To solve this problem, Ding et al. (2018) proposed a robust two-stage algorithm for local community detection (RTLCD). Specifically, RTLCD is divided into two stages: core detection and community expansion. (a) In the first stage, the method starts with an initial query node and explores the core nodes in the network with high clustering tendency by breadth-first search; (b) In the second stage, the core nodes are used as query nodes to find the community by community expansion methods. However, the limitation of the method is to only consider structural quality in the core detection stage which makes it impossible to be extended to attributed graphs. For the second stage, RTLCD believes that other nodes that are connected to the core node and have high structural quality should be added to the community. However, for the attributed graph, community members should also satisfy the condition of having similar attributes to the query node, therefore exploring community members simply by the quality of the structure is no longer suitable for local community detection on the attributed graph.

Although it is promising to replace the query node with a core node, it still faces the following two challenges.

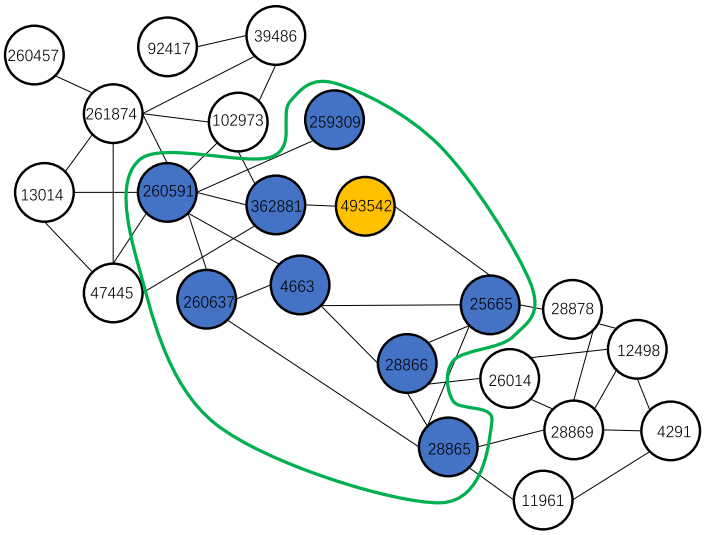
(1) How to integrate attribute information into the seed replacement stage? Different from RTLCD, in terms of attributes, the core nodes need to be as similar as possible to the other members of the community , as well as similar to the attributes of the query nodes.
(2) How to develop a seed replacement strategy for all attribute types? Different types of attributes require different ways to measure importance. It is meant to develop a seed replacement strategy suitable for multiple attribute types.

Towards this end, we develop a two-stage community search method SRRW, a novel algorithm that provides a comprehensive approach to joint seed substitution and random walk of multi-type attributed graphs. Specifically, SRRW is divided into two stages: seed replacement and joint random walk community search. In the first stage, we first reconstruct the attributed graph into an augmented graph, and then we propose dynamic local clustering coefficients and attribute cluster central membership matrix based on the attributed and augmented graph, respectively, and finally, the query nodes are updated through a seed replacement strategy. In the second stage, we combine structure and attribute information via enabling walkers to jump on both the attributed and augmented graph. After obtaining the node importance ranking, the community is captured by minimizing the parallel conductance.

The main contributions are summarized as follows:

- We propose cluster-center membership coefficient and dynamic local clustering coefficient inspired by the augmented graph and local clustering coefficient.
- To explore community in graph with multiple types of attributes and enhance the robustness of the method, we have designed a new two-stage community search method based on seed replacement and joint random walk.
- We perform extensive experiments on a variety of real-life datasets and synthetic datasets. The results demonstrate the effectiveness and efficiency of our method.

The remainder of this paper is organized as follows. Section 2 reviews related studies. Sections 3 to 5 introduce the proposed SRRW method. We verify our method on several real datasets and synthetic datasets, and the experimental results demonstrate the effectiveness of our model in Sect. 6. Section 7 shows our conclusions and describes some insights for future research.



Number of nodes	22
Number of edges	76
Number of communities	2
Average degree of nodes	3.455

Fig. 1 Community search results (with PRN) for the query node from boundary region. The yellow node is the initial query node (boundary node) and the blue nodes are the community nodes found by the PRN. Obviously, not these blue nodes are from the same community

2 Related work

In this section, we review the existing approaches that are most relevant to our method, in particular the community search over simple graphs, community search over attributed graphs, and random-walk-based community search. Then we briefly explain the differences with our method.

2.1 Community search over simple graphs

At the early stage of the study of community search models, the definition of community varies among different studies, cohesive subgraphs like maximal cliques (Cheng et al. 2011a), k-core (Cheng et al. 2011), k-truss (Akbas and Zhao 2017; Wang and Cheng 2012), etc. form the basis of modeling communities. In particular, the k-core-based community search methods return the community in which the degree of every vertex is no less than k (Cui et al. 2014). Sozio and Gionis (2010) motivate a measure of density based on a minimum degree(k-core) and distance constraints, and develop an optimum greedy algorithm for this measure. However, it is well known that the k-core community is not guaranteed to be cohesive. In other words, k-core only requires that the degree of nodes in the community is not less than k, which cannot indicate that the community has the characteristics of high cohesion. To ensure the cohesiveness of the retrieved community, clique (Yuan et al. 2017) and k-truss have also been considered for community search. However, as the clique model is too restrictive, some relaxed variants have been investigated (Cui et al. 2013).

However, many of the aforementioned methods suffer from the query-bias issues that detection results contain error nodes if the query nodes are from the community boundary region. To solve the seed-dependent problem, Ding et al. (2018) propose RTLCD based on core detecting and community extension. The core detecting stage replaces the seed with the core member of the target community, the community extension stage takes the detected community core member as an initial community and extends the community based on relation strength. Bian et al. (2020) propose an effective amplified topology potential (ATP) algorithm to detect core nodes of the target communities w.r.t original query nodes.

Although seed replacement can avoid seed dependency, because of the loss of attribute information, ARLCD and ATP cannot locate a community with similar attributes, that is, members in the community have the same semantic attributes.

2.2 Community search over attributed graphs

Except for simple graphs, community search has also been studied for more complex graphs, such as community search over attributed graphs (Zhao et al. 2022; Li et al. 2022), geo-social graphs (Luo et al. 2020; Chen et al. 2018), and so on. In particular, Fang et al. (2017) and Fang et al. (2016) propose the ACQ algorithm to find subgraphs satisfying structural and keyword cohesiveness. Huang and Lakshmanan (2017) also explore attribute-driven CS in terms of k-truss. Most of these works study keyword-based CS that take a set of keywords or a query vertex as input and return a subgraph as the community that has the best match with the given set of query keywords.

These works only consider the attributes of networks and ignore the type of attributes. For example, ACQ and ATC only consider categorical attributes. The categorical attributes can

only indicate whether the node has the attributes, and cannot give the strength of the node's preference for the attributes. However, many real-world networks use attribute similarity (or other numerical attributes) as node attributes, e.g., social networks, protein networks, etc. For these networks, ACQ and ATC treat numerical attributes as categorical attributes, which makes the communities found by these methods deviate from the benchmark. In addition, these works do not aim to solve the seed-dependent problem. However, the experimental results show that when the query node is located in the boundary area of the community, the performance of ACQ and ATC has declined. This shows that the study of effective seed replacement strategy is conducive to enhancing the robustness of the method.

2.3 Random-walk based community search

Random walk-based methods have also been routinely applied to search local communities in a network. A walker explores the network following the topological transitions. The node visiting probability is usually utilized to determine the detection results. For instance, Yin et al. (2017) propose a motif-based random walk model and search node sets with minimal motif conductance. MWC (Bian et al. 2017) sends multiple walkers to explore the network to alleviate the query-bias problem. Note that all aforementioned methods are only designed for simple graphs, and neglect the effect of attributes. There are some methods like PRN that suffer from the seed dependent issue that detection results contain false nodes if the query node is from community boundary region.

Community search based on random walk is also widely used in attributed graphs, which aims to mine communities with tightly connected structures and node attributes with the most similar attributes possible. Based on this idea, most methods first obtain the edge weights by similarity calculation (attribute distance or attribute similarity) and then perform random walk to locate a local community. For example, Hsu et al. (2017) propose an unsupervised learning framework, AttriRank, to improve the reliability of node importance ranking. However, attribute similarity is used as the edge weight which results in the loss of direct relationship between attributes and nodes.

3 Preliminaries

Let $G = (V, F, \mathbf{A}, \mathbf{Q})$ be an undirected node-attributed network, where $V = \{v_1, v_2, \ldots, v_n\}$ is the set of nodes, connected by an undirected network adjacency matrix denoted as $\mathbf{A}_{n\times n}$. For each pair of nodes v_i and v_j, if there is no link between them, A_{ij} would be 0, otherwise, A_{ij} would be 1. $F = \{f_1, f_2, \ldots, f_m\}$ is the set of attributes. We use the matrix $\mathbf{Q}_{n\times m}$ to collect all the node attributes. For each pair of nodes v_i and attributes f_j, if v_i has the attribute f_j, $Q_{ij} = 1$; otherwise, $Q_{ij} = 0$.

Given a seed node v_{seed} and an undirected node-attribute graph G, our goal is to find a community D_{seed}, such that D_{seed} is a connected component containing v_{seed}. The target community D_{seed} is expected to have members with structure cohesion and attribute homogeneous. In addition, D_{seed} should be as similar to ground-truth C_{gt} as possible. Table 1 lists some important notations used in this paper.

3.1 Local clustering coefficient (LCC)

It is possible and meaningful to find some measures to analyze the clustering tendency of a given node. As is known to all, nodes in a more central region of the cluster usually own a higher clustering tendency than others. Conversely, the larger the clustering coefficient, the more possibly the nodes are in the core community. Thus, We follow LCC (Nascimento 2014) to evaluate the clustering tendency of nodes as defined:

$$\text{LCC}(v_i) = \frac{2 \times \sum_{j,k\in N(v_i)} A_{jk}}{k_i \times (k_i - 1)} \tag{1}$$

where $N(v_i)$ is the neighbors set of node v_i, and k_i is the degree of v_i. The value of $\text{LCC}(v_i)$ ranges from 0 to 1. The value 0 means there is no clustering feature between v_i and its neighbors. The value 1 means that they are completed linked. A higher $\text{LCC}(v_i)$ indicates a higher local clustering tendency of node v_i.

3.2 Random Walk with Restart (RWR)

RWR is a general random walk model for topological networks and can be further customized into different variations. In RWR, at each time point, the random walker explores the network based on topological transitions with $\alpha (0 < \alpha < 1)$ probability and jumps back to the query node with probability $1-\alpha$. The restart strategy enables RWR to obtain proximities of all nodes to the query node. It defines as:

$$\mathbf{r}^{t+1} = \alpha \times \hat{\mathbf{A}} \times \mathbf{r}^{t} + (1 - \alpha) \times \mathbf{q} \tag{2}$$

where $\mathbf{q}$ is the restart vector that contains the element 1 on the position that corresponds to the seed node and zeros elsewhere. $\mathbf{r}^{t+1}$ is the node visiting probability vector at time t. A higher value in the $\mathbf{r}^{t+1}$ indicates that the node is more intimate to the target node.

Table 1 Notations and meanings

Notation	Definition
$\widehat{\mathbf{A}}$	Row normalization matrix of $\mathbf{A}$
k_i	The degree of vertex v_i
$\mathbf{D}^v$, $\mathbf{D}^a$	Diagonal matrix of nodes and attributes
S	Node attribute matrix
R	Transition probability matrix
α,β	Restart factor
r^t	Node visit. pro. vec. of the walker in G at t
q	One-hot vector with only one value-1 entry for v_{seed}
t_{find}	The number of iterations to find the replacement node
$N(v_i)$	Neighbor node set of node v_i

4 The proposed algorithm

Existing local community detection methods usually ignore the following two key issues: on the one hand, users usually randomly choose query nodes, and the nodes at the community boundary may be adopted as the starting nodes for local community detection. A low-quality query node can lead to an incorrect local community result; on the other hand, researchers usually employ metrics such as attribute similarity to determine the semantic relationships between nodes on an edge. However, attribute should be considered more as an another type node rather than as edge weight.

To address these two problems, we propose a two-stage community search method with seed replacement and joint random walk as shown in Fig. 2.

The first stage consists of three steps as follows: first, we develop an index to evaluate the quality of the node structure, i.e. dynamic local clustering coefficient (DLCC); second, we construct the augmented graph to calculate the cluster center membership matrix (CCMM), and finally we propose a seed replacement process based on the results of steps 1 and 2.

The second stage based on the joint random walk is divided into two steps as follows: first, joint random walks are performed on attributed graph and augmented graph; second, we propose parallel conductance value and combine it with joint random walk to find a community.

In the following, we present the proposed SRRW method based on the above two stages.

4.1 The seed replacement stage

4.1.1 Dynamic local clustering coefficient

As previously mentioned, traditional measures of clustering tendency only take into account the closeness of a given node's neighbors and omit the effect of the node's own degree, which leads to erroneous amplification of the clustering tendency of a node with a small degree and closely connected neighbors. To solve this problem, we propose DLCC as follows:

$$\text{DLCC}(v_i) = \sigma(k(v_i)) \times \left(\frac{2 \times \sum_{v_j, v_k \in N(v_i)} A_{ik}}{k(v_i) \times (k(v_i) - 1)} \right) \tag{3}$$

$$\sigma(x) = \frac{1}{\max_{0<j\leq|V|}(d(v_j))} x, \tag{4}$$

where $\sigma(x)$ is based on the maximum degree in the network, its purpose is to assign the importance of nodes with different degrees to (0, 1). For DLCC, the value 0 means there is no clustering feature between the node and its neighbors. The value 1 means that they are completed linked.

Table 2 shows the LCC and DLCC of node in Fig. 3. v_7 and v_9 are the nodes with the best clustering tendency based on LCC value. In terms of DLCC, v_3 is the best node, which is in line with the real scenario.

4.1.2 The augmented graph construction method and CCMM

Existing methods mainly take attribute similarity as the edge weight between nodes. The ownership of attributes for a particular node can be naturally taken as an interaction between these two heterogeneous sources. Thus it is more reasonable to regard attribute as another type of node than edge weight as nodes with different attributes share functionality similarity. Meanwhile, nodes of similar attributes reflect similar attribute subspace as well.

Inspired by the above insights, Zhe et al. (2019) have proposed an augmented graph construction method based on attribute centers, which first finds the attribute centers via k-means clustering and then connects them to nodes to construct an augmented graph with two types of nodes. However, one single attribute center may cover incomplete attribute profile. For example, in a social network, a user may like both ball sports and athletics, one of which would be ignored if the user is only associated with a single attribute center. Instead

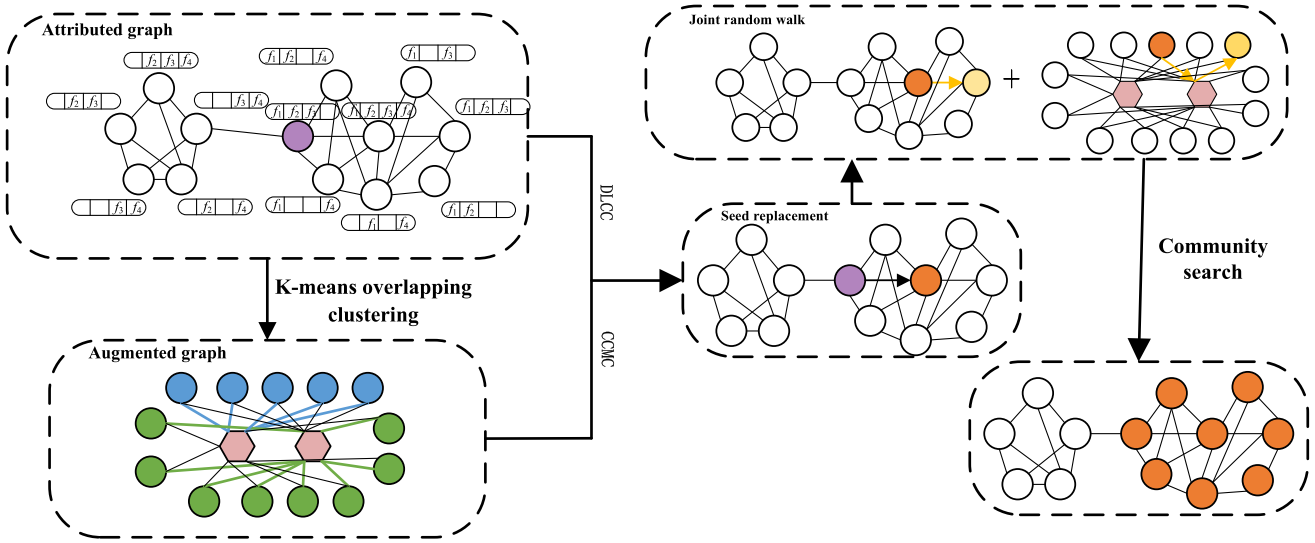


Fig. 2 An illustration of SRRW framework. Firstly, the augmented graph are constructed by k-means attribute overlapping clustering method, DLCC and CCMM are calculated based on the two graph respectively; secondly, the core node are found by seed replacement strategy, and finally the core node are used as query node to execute joint random walk to locate community

Table 2 LCC and DLCC

Node	LCC (v_i)	DLCC (v_i)
v_1	0.500	0.334
v_2	0.833	0.566
v_3	0.667	0.667
v_4	0.333	0.222
v_5	0.833	0.556
v_6	0.600	0.500
v_7	1	0.333
v_8	0.667	0.445
v_9	1	0.500

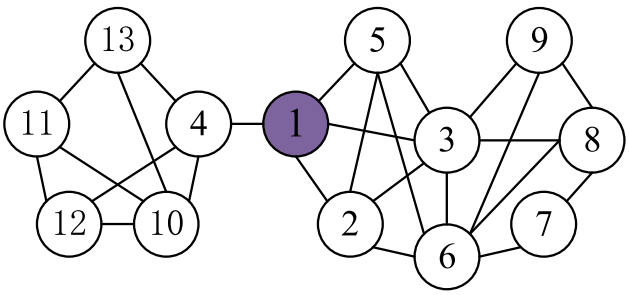


Fig. 3 Sample graph

of directly connecting one node with one attribute center, we propose an augmented graph construction method on the basis of overlapping attribute centers. The method assigns one node to multiple attribute centers with two advantages. For one thing, our construction method can be applied not only to a graph with categorical attributes but also tailored for all types of attributes as long as the attributes are available for overlapping clustering. Our method is also flexible since all kinds of center-based attribute clustering algorithms can be easily adopted (here we use a k-means overlapping clustering method (Liu et al. 2020a)). For another, we convert the relationship between nodes and their attributes into the relationship between a node and attribute centers, which can effectively reduce the time complexity of constructing an augmented graph.

To indicate the strength of the belongingness relationship between each vertex and its nearest attribute center, we use attribute distance to initialize the weight of a belongingness edge. For example, we can use Euclidean distance if the k-means algorithm is performed to cluster attribute values. P_{ij} is defined as the attribute distance between node v_i and attribute centre c_j. Let $\mathbf{P}_{n\times k}$ be the node attribute center interaction matrix, P_{ij} represents the strength of the relationship between node v_i and attribute center c_j, we compute P_{ij} as follows:

$$P_{ij} = \text{soft max}\left(T_{ij} \times \frac{1}{d(v_i, c_j)}\right) \tag{5}$$

where T_{ij} is based on the relationship between node v_i and attribute center c_j. If there is no link between them, T_{ij} would be 0, otherwise, T_{ij} would be 1. $d(v_i, c_j)$ represents the Euclidean distance between node v_i and attribute center c_j.

The ith row of $\mathbf{P}$ indicates the affiliation information of node v_i with k attribute centers. A higher value of $\mathbf{P}_{ij}$ shows that node v_i is more probable to belong to attribute center c_j. In other words, $\mathbf{P}_i$ represents the attribute center

affiliation distribution of node v_i. Intuitively, if the attribute center affiliation distributions of two nodes are similar, the more likely the two nodes contain similar side information. According to this insight, we define the attribute center similarity of the nodes as $sim(v_i, v_j) = sim(P(i,), P(j,))$ and store their values in CCMM(v_i, v_j) and CCMM(v_j, v_i). The CCMM(v_i, v_j) is replaced by CM(v_i, v_j) in the corresponding position in the later text. Since in the node replacement stage, we are more interested in whether the candidate node is similar to the query node, at this time we can fix a row or column in the matrix CM as the query node v_{seed}. We use $CM(v_i, v_{\text{seed}})$ as the attribute evaluation index of node v_i, and a larger $CM(v_i, v_{\text{seed}})$ indicates that node v_i is more similar to v_{seed} in terms of attributes.

4.1.3 Seed replacement procedure

To find core nodes, we propose the seed replacement stage which fulfills the following two conditions. First, to avoid detecting core members of unrelated communities, the seed replacement stage should ensure the detected core member is tightly related to the seed node; second, the seed replacement stage should be able to detect a core member of the target community from any seed nodes.

To fulfill the first condition, we propose a seed replacement stage based on dynamic local clustering coefficient and attribute similarity. At each iteration, the approach replaces the seed node with its most similar and more influential neighbor. We keep the number of iterations between 3 to 5 times to avoid replacing nodes that are too further away from the given query node, which ensures that the given node must be in the community found by the algorithm. To fulfill the second condition, we develop the seed replacement stage as a reversed influence spreading method. In each iteration, the method replaces the seed with a node which is closer to the core of the target community. Thus, the method can form a replacement path from any seed to the core member of the target community.

In the seed replacement process, we first put the neighbors of the initial seed node into the candidate node set $v_{\text{candidate}}$, after which the structure evaluation index and attribute evaluation index of the node are obtained by Eqs. (3), (4) and (5). In the selection expectation of the replacement node, we expect the replacement node to have a better structure quality than the query node while maintaining similar attributes to the initial query node. Therefore, we require the structure quality of the replacement node to be greater than that of the query node in step 8 and the similarity between the replacement node and the initial query node to exceed a threshold θ in step 9. Seed replacement pseudo-code is further summarized in Algorithm 1.

To effectively integrate attribute information in random walk, we first propose the joint random walk method. The core idea is to perform a joint random walk on the augmented graph to capture nodes that are highly similar to the query node. The following sections will introduce the joint random walk and community search method respectively.

Algorithm 1 Seed replacement

Require: Attributed Graph G; seed node v_{seed}; similarity threshold θ; iteration number T; adjacency matrix $\mathbf{A}$; node-attribute matrix $\mathbf{Q}$;
Ensure: The replacement node $v_{newseed}$; the number of iterations to find replacement node t_{find};
1: Construct augmented graph and node cluster center membership matrix $\mathbf{M}$
2: Find $N(v_{seed})$ according to the adjacency matrix,
3: $v_{candaidate} = v_{seed}, t_{find} = 0, N_{all} = N(v_{seed})$
4: According to formula(3)(4), get DLCC(v_i),
5: CM(v_{seed},v_i) for each node in N(v_{seed}), and get DLCC(v_{seed})
6: **while** true **do**
7: $N_{tem} = \varnothing$, $t_{find} += 1$
8: **for** $v_i \in N_{all}$ **do**
9: $N_{tem} = N_{tem} \cup N(v_i)$
10: **if** $DLCC(v_i) > DLCC(v_{seed})$ **then**
11: **if** $CM(v_i, v_{seed}) > \theta$ **then**
12: $v_{candidate} = v_i$
13: **end if**
14: **end if**
15: **end for**
16: **if** $t_{find} > T$ **then**
17: break
18: **else**
19: **if** $v_{candidate} == null$ **then**
20: $N_{all} = N_{tem}$
21: **else**
22: $N_{all} = N_{tem} \cup (N(v_{candidate}) - N(v_{seed}))$
23: **end if**
24: **end if**
25: **end while**
26: **return** $v_{newseed}, t_{find}$

4.2 The joint random walk community search stage

4.2.1 Joint random walk

In this section, we perform a joint random walk on the augmented graph. A walker in the joint random walk is jointly influenced by the structure and attribute information. The proposed walking mechanism can propel the random walks more diverse.

Let $\widehat{\mathbf{P}}_{n\times n} = \mathbf{P}_{n\times k}\mathbf{P}^T_{k\times n}$ represents the node-attribute center-node transition probability matrix. $\widehat{\mathbf{P}}_{ij}$ is the possibility of transferring from node v_i to v_j through several attribute centers. This method increases the importance of nodes whose attributes are similar to the seed node. To balance the elements in $\widehat{\mathbf{A}}_{n\times n}$ and $\widehat{\mathbf{P}}_{n\times n}$, we use β to adjust the importance between them, as in:

$$\mathbf{R} = \beta\widehat{\mathbf{A}} + (1-\beta)\widehat{\mathbf{P}}. \tag{6}$$

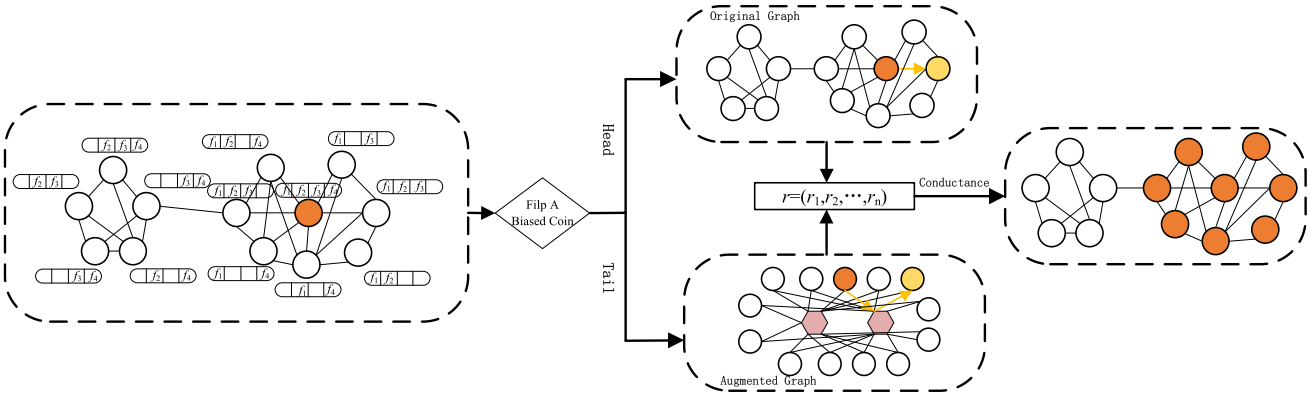


Fig. 4 Framework of community search based on joint random walk. Assuming the orange node in the attributed graph as the query node, a biased coin is tossed and the walker explores the structural information on the original graph if heads are facing up, and the walker explores the attribute information in the augmented graph if tails are facing up. Finally, the community is located by minimizing the conductance value

Then, we apply the restart strategy for joint random walk in updating visiting probability vectors. For a walker, we have:

$$\mathbf{r}^{t+1} = \alpha \times \mathbf{R} \times \mathbf{r}^{t} + (1-\alpha) \times \mathbf{q}. \tag{7}$$

The proposed joint random walk would jump among all these $(n+k)$ nodes. As illustrated in Fig. 4, Assume that we have jumped from an orange node v_i. To determine the next transition, we flip a biased coin, if it yields head, then we walk one step on the original graph G. If it turns tail, then we walk two steps on the augmented graph.

The key difference between the joint random walk and the random walk with restart is the addition of the attribute centers node. RWR spread the influence of the query node to the entire graph through the topology structure, and returns the tightly connected nodes. However, the target community in attributed community search needs to satisfy the structure cohesiveness and attribute similarity respectively. A joint random walk can transfer the influence of seed nodes to other nodes through the attribute center. Therefore, a joint random walk can improve the intimacy between nodes with similar attribute centers. Experiments prove that this method can improve the accuracy of the community results.

4.2.2 Parallel conductance value

Traditional conductance values are often used to capture local communities, such as PRN. PRN scans the ranking list to find the subset of top-ranked nodes that minimizes the conductance of the local community. However, the classical conductance value does not consider attribute information. To solve this problem, we propose parallel conductance values.

Let $\mathbf{W}_{n\times n}$ be the node attribute similarity matrix, where W_{ij} represents the attribute similarity of nodes v_i and v_j. For each pair of nodes, we have:

$$W_{ij} = \frac{\|Q_i \odot Q_j\|_0}{\|Q_i + Q_j\|_0}, \tag{8}$$

where $\odot$ represents the elementwise product, $||Q_i||$ is the 0-norm of the vector Q_i, that is, the number of non-zero elements in the Q_i.

The parallel cut of the fusion structure and attributes is defined as follows:

$$\text{parallel\_cut}(D) = \sum_{i\in D} \frac{\sum_{j\notin D} A_{ij} + W_{ij}}{\sum_{j\in D} A_{ij} + W_{ij}}. \tag{9}$$

We define parallel conductance combining structure and attributes as follows:

$$\text{Con}(D) = \frac{\text{parallel\_cut}(D)}{\text{vol}(D)}. \tag{10}$$

Algorithm 2 summarizes the pseudo-code of attributed community search based on a joint random walk. Firstly, we add the t find order neighbors of the seed node into the initial community, denoted as D_{intial}. t_{find} is the number of iterations when finding a replacement node. The method ensures that the original query node must be included in the resulting community. Secondly, To find the local community contains seed node in network G, let $\{s_i\}$ represent the list of nodes sorted in descending order by its influence score. Then for each s_i, we compute the conductance of the subgraph induced by node-set $D_{\text{intial}} \cup \{s_i\}$. The node set with the smallest conductance will be returned as the local community.

Algorithm 2 Community search based on joint random walk

Require: Attributed Graph G; Augmented Graph $\widehat{G}$;
Query node $v_{newseed}$; t_{find}; Number of iterations $iter$
Ensure: Output Community D;
Construct matrix $\mathbf{P}$ according to augmented graph
$\widehat{G}$ and equation 5
Construct the transition matrix $\mathbf{R}$ according to equation 6
while $t < iter$ **do**
  $\mathbf{r}^{(t+1)} = \alpha \times \mathbf{R} \times \mathbf{r}^t + (1+\alpha) \times \mathbf{q}$
  $t = t+1, r^t = r^{(t+1)}$
end while
Find $N(v_{seed})$ according to the adjacency matrix,
$v_{candaidate} = v_{seed}, t_{find} = 0, N_{all} = N(v_{seed})$
Store the t_{find} order neighbors of the seed node as the initial community in D_{intial}
$D_{candidate} \leftarrow N(D_{intial}), C_{onintial} = Con(D_{intial})$
for $v_i \in N_{all}$ **do**
  $Con = Con(D_{intial} \cup v_{candidate})$;
  if $Con > Con(D_{intial}) and D_{candidate} == \varnothing$ **then**
    return D_{intial}
  else
    if $Con > Con(D_{intial}) and D_{candidate} \neq \varnothing$ **then**
      continute
    else
      $D_{intial} = D_{intial} \cup \{v_{candidate}\}$
      $D_{candidate} = D_{candidate} \cup N(v_{candidate})$
    end if
  end if
end for
return D

5 Example and reasonableness

In this section, we introduce the seed replacement strategy of SRRW through two stages of calculating DLCC and the seed replacement process. We sampled a partial dataset containing two benchmark communities from the DBLP dataset. This dataset contains 22 nodes with co-authorship relationships between nodes as authors. The authors' attributes are bags of words represented by keywords.

Stage 1: Calculate the DLCC of nodes

First, we give the DLCC of all nodes in the network as shown in Table 3. We present the computation process of DLCC using node 260,591 as a case study. As shown in Fig. 1, the maximum degree in the network is 8, then DLCC(260,591) $= 1/8 \times 8 \times (2 \times 7)/(8 \times 7) = 1/4$.

Stage 2: Seed replacement process

Suppose the given query node is 493,542, with DLCC(493,542) = 0.

When $t = 1$, the DLCC values of its neighbors are DLCC(25665) = 0.063 and DLCC (362,881) = 0.167, respectively. after calculation, sim(M(25,665), M(493,542)) = 0.283 and sim(M(362,881),M(493,542)) = 0.533. We will replace 493,542 with 362,881 according to Algorithm 1.

When $t = 2$, the DLCC values of the neighbors of node 362,881 are DLCC(102,973) = 0.125, DLCC(260,591) = 0.25, and DLCC(47,445) = 0.333. Meanwhile, sim(M(102,973), M(362,881)) = 0.603, sim(M(260591), M(362,881)) = 0.588 and sim(M(47,445), M(362,881)) = 0.681. Therefore, node 362,881 is replaced with node 47,445 according to Algorithm 1.

When $t = 3$, the DLCC value of the neighbors of node 47,445 as follows: DLCC(13,014) = 0.375, however sim(M(13,014), M(47,445)) = 0.533. Since $0.533 < 0.681$, node 47,445 is kept unchanged. The replacement path of the node is shown in Figure 5.

Stage 3: Joint random walk

We take node 47,445 as the query node and the joint random score (JRS) of the nodes are shown in Table 4. Figure 6 shows the results of SRRW on the DBLP network. Compared to Fig. 1, the community in Fig. 6 is clearly closer to the benchmark community. Intuitively, this is due to the replacement of the boundary node 493,542 with the core node 47,445 in the seed replacement phase. Since node 259,309 has fewer neighbors, it can only be accessed by walkers through one path. This structural deficiency leads to a low score for similar nodes. However, node 259,309 has a slightly higher score than other community nodes, and setting a lower threshold still allows it to be included in the community (e.g., 0.025).

Table 3 The calculated DLCC scores for the nodes of partial DBLP network

Node	DLCC	Node	DLCC	Node	DLCC
261,874	0.25	13,014	0.375	47,445	0.333
260,591	0.25	260,637	0.125	4663	0.167
362,881	0.167	493,542	0	259,309	0
102,973	0.125	39,486	0.125	92,417	0
25,665	0.063	28,866	0.167	28,865	0.063
26,014	0.125	28,878	0.125	12,498	0.25
28,869	0.188	11,961	0	4291	0.125
260,457	0				

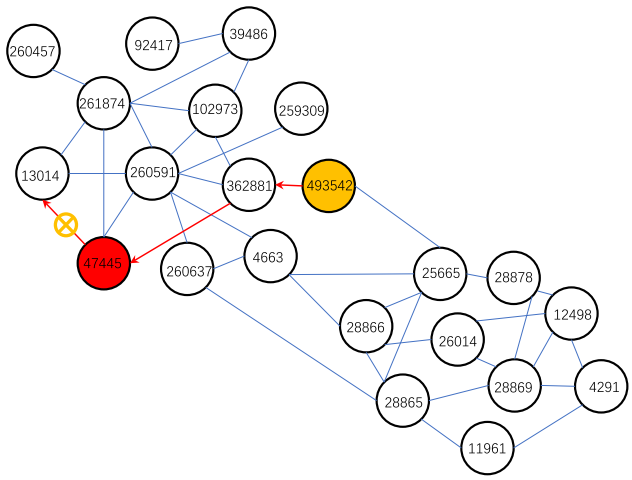


Fig. 5 Node replacement path graph

Table 4 The calculated JRS for the nodes of partial DBLP network

Node	JRS	Node	JRS	Node	JRS
261,874	**0.056**	**13,014**	**0.063**	47,445	0.150
260,591	**0.052**	**260,637**	**0.042**	**4663**	**0.040**
362,881	**0.058**	**493,542**	**0.0430**	259,309	0.025
102,973	**0.049**	**39,486**	**0.047**	**92,417**	**0.046**
25,665	0.022	28866	0.021	28,865	0.021
26,014	0.018	28878	0.018	12,498	0.018
28,869	0.018	11961	0.018	4291	0.018
260,457	**0.05**				

Bold values represent a higher visiting probability (exceeds a predefined threshold)

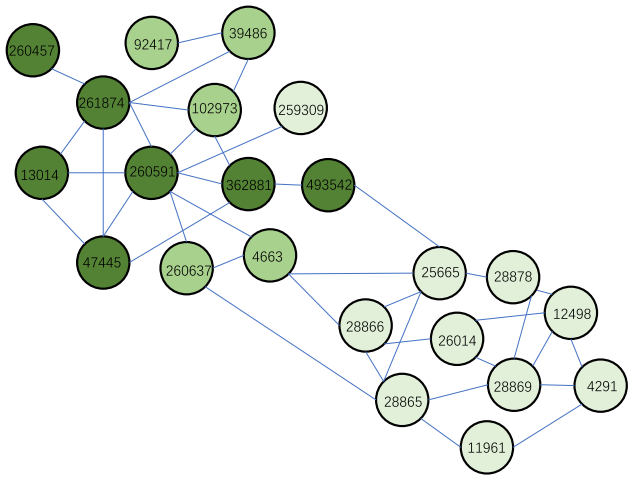


Fig. 6 Results of SRRW for given query node 47,445. The nodes are colored according to their JRS generated by SRRW. Darker color represents a higher visiting probability

6 Experimental results

In this section, we conduct experiments to answer the following research questions:

RQ1: How do hyper-parameters (k and α, β) in SRRW impact community search performance?
RQ2: How does our proposed SRRW model perform compared with state-of-the-art community search approaches?
RQ3: How does SRRW benefit from its components (i.e., seed replacement and joint random walk)?

All algorithms are coded in python3.8, and all the experiments are implemented on a computer with a 3.4 GHz CPU and 32 GB memory. We first present datasets, evaluations, and comparison methods, followed by answering the above three research questions.

Table 5 Descriptive statistics of real-world dataset

Dataset	$\|V\|$	$\|E\|$	$\|F\|$	k_{avg}	N.o.c
CORA	2708	5428	1432	3.797	7
IMDB	35,389	79,642	30,789	4.500	8
SINNET	55,373	102,567	1232	3.320	11
DBLP	317,080	1,049,866	50,337	3.9	5000

Table 6 Descriptive statistics of synthesis dataset

Dataset	$\|V\|$	$\|E\|$	$\|F\|$	k_{avg}	N.o.c
LFR-2	200,000	2,025,600	20,000	20.256	1360
LFR-5	500,000	7,366,409	50,000	29.466	3480

6.1 Datasets

We conduct extensive experiments to evaluate the performance of the proposed method using a variety of real-world networks and synthetic networks. We apply our model to four public accessible datasets for community search. The statistics of the datasets are summarized in Table 5. N.o.c is the number of community.

CORA is a citation network. Nodes represent the publications. Edges represent the reference relationship among publications. Attributes of nodes are defined as the keywords of the publications.

IMDB is extracted from an internet movie database. Edges indicate that the two movies are directed by the same director and have common actors. Attributes of nodes are the Bag-of-words of the directors and actors.

SINANET is a microblog user relationship network extracted from the Sina-microblog website. Each vertex represents a user and each edge represents a relationship. Attributes are extracted by the LDA topic model that represents user's topic distribution.

DBLP is a co-authorship dataset, nodes are authors and edges indicate co-authorship between authors. Authors are divided into 5000 communities. Community labels are assigned to authors based on the conference they contributed to. The authors' attributes are bags of words represented by keywords.

For synthetic networks, we use the LFR to generate two networks, the statistics of these two networks are shown in Table 6. When assigning attributes, we divide nodes according to the similar attributes within the community and the different attributes outside the community.

Table 7 LFR parameters and meanings

Parameter	Meaning
N	Number of nodes
$\overline{k}$	Average degrees
$\max_k$	Maximum degree
μ	Mixing parameters
τ_1	Negative index of degree series
τ_2	Negative index of community size distribution
$\min_c$	Minimal community Size
$\max_c$	Maximum community Size
O_n	Number of overlapping nodes
O_m	Number of communities to which overlapping nodes belong

The parameter settings for LFR-2 and LFR-5 are shown below. The meanings of the parameters are summarized in Table 7.

LFR-2: $N = 200{,}000$, $\overline{k} = 10$, $\max_k = 50$, $\mu = 0.1$, $\tau_1 = 2$, $\tau_2 = 1$, $\min_c = 1000$, $\max_c = 2000$, $O_n = 0$, $O_m = 0$.

LFR-5: $N = 500{,}000$, $\overline{k} = 10$, $\max_k = 80$, $\mu = 0.2$, $\tau_1 = 3$, $\tau_2 = 2$,$\min_c = 2000$, $\max_c = 4000$, $O_n = 0$, $O_m = 0$.

6.2 Evaluations

We use recall, precision, F_1, local modularity (Q_l), and node coverage rate (NCR) to evaluate the performance of detected local communities. They are defined as follows.

$$\text{recall} = \frac{|C_F \cap C_T|}{|C_T|} \tag{11}$$

$$\text{precision} = \frac{|C_F \cap C_T|}{|C_F|} \tag{12}$$

$$F_1 = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{13}$$

where C_F is the community detected by the algorithm, and C_T is the real community to which the given node belongs. The recall represents the ratio of the number of detected nodes that belong to the real community to the number of nodes in C_T. precision represents the proportion of the correctly detected nodes in C_F. Moreover, F_1 is the harmonic mean of recall and precision. The values of recall, precision, and F_1 are between 0 and 1, and a larger value implies a better algorithm performance.

The definition of local Q_l is denoted as:

$$Q_l = \frac{k_{\text{in}}}{k_{\text{in}} + k_{\text{out}}} \tag{14}$$

Where k_{in} represents the number of edges between the boundary nodes and other nodes in the local community, and k_{out} is the number of edges between the boundary nodes and the nodes outside the local community.

To show the performance of the seed replacement component in SRRW, we suggest NCR represent the proportion of valid seeds in all the seeds used by an algorithm.

$$\text{NCR} = \frac{|V_{\text{valid}}|}{|V_{\text{used}}|}. \tag{15}$$

6.3 Comparison methods

Here we mainly validate whether our new SRRW framework is competitive with or performs better than the existing community search methods, particularly in the realm of attributed community search models. We compare SRRW to three categories of methods as follows. First, to study how DLCC improves the effectiveness of seed replacement. We replace DLCC with the LCC and the LCC improved by using sigmoid. These two methods are denoted as SRRW-L and SRRW-S respectively. Second, to analyze how does SRRW benefits from its components. We remove the seed replacement and replace the augmented graph with a bipartite graph respectively, and denote the two methods as SRRW-NSC and SRRW-BG. Third, to verify the effectiveness of SRRW, we select three methods using only topology information, i.e., RTLCD, TSB, and PRN. We thoroughly evaluate SRRW on attributed community quality by comparing SRRW with two state-of-the-art baseline methods, i.e., ACQ, VAC.

RTLCD Ding et al. (2018): a robust two-stage local community detection algorithm based on core detecting and community extension.

TSB Liu and Xia (2020): this method is a local community detection method based on breadth first search, transfer similarity, and local clustering coefficient.

PRN Tong et al. (2006): this method uses conductance value and random walk for community search.

ACQ Fang et al. (2016): this method aims to find an attributed community for a given query node and a set of query keywords. Specifically, the commu-

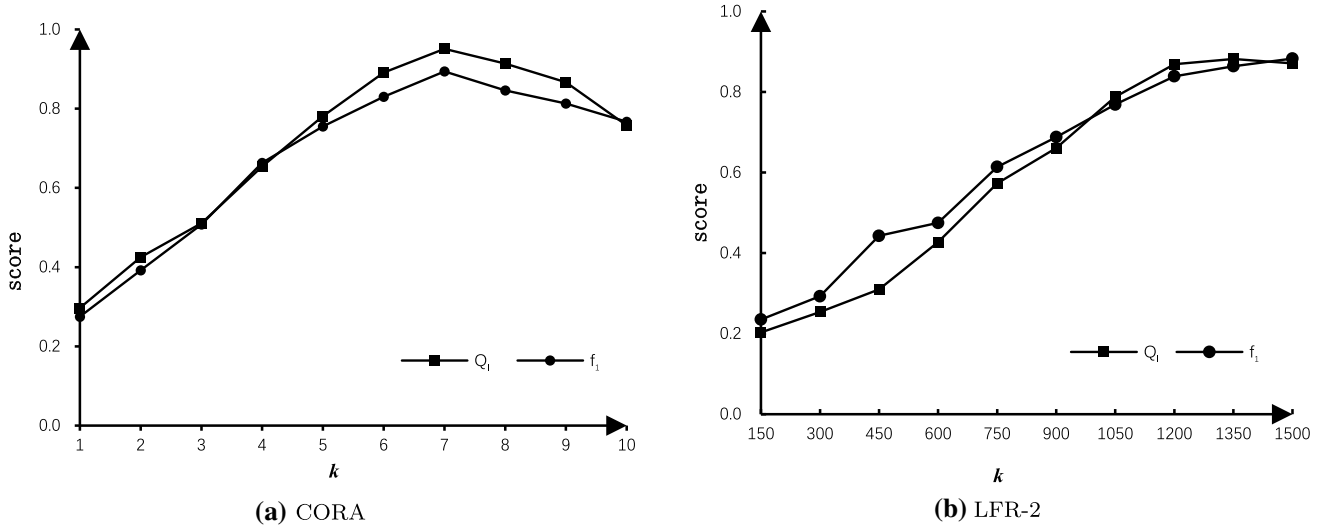


Fig. 7 Performance w.r.t. different k on CORA and LFR-2

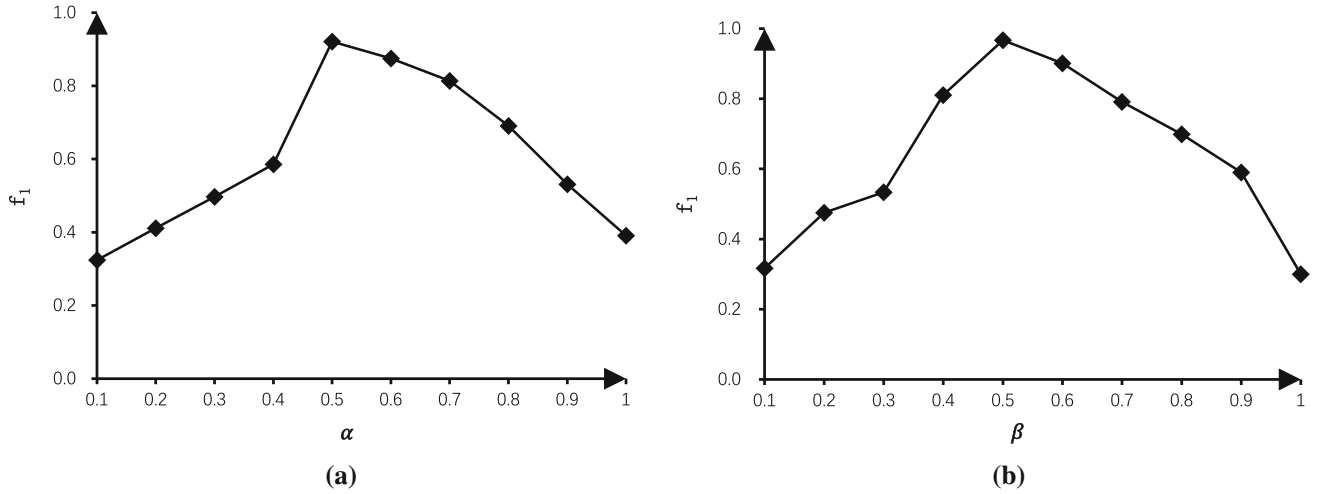


Fig. 8 Performance w.r.t. different α and β on CORA

nity is a k-core and the number of common query keywords is maximized for all vertices in the subgraph.

VAC Liu et al. (2020): this method proposes a vertex-centric attributed community that takes into account both spatial information and keywords associated with vertices.

6.4 Parameter sensitivity analysis (RQ1)

SRRW has three parameters, k, the parameter in overlapping clusters. α and β, the parameters in the joint random walk. We respectively set the default value of k, α (or β) to be the number of ground-truth communities in the current experimental dataset and 0.5. When testing one of these parameters, the other two parameters are set to default values. For each dataset, we randomly select 100 nodes as the query nodes. The average values of F_1 and Q_l in the 100 nodes of the network are the final experimental results. Because the experimental results on all datasets are similar, we only show the average experimental results on CORA and LFR-2 as shown in Fig. 7a, b.

Figure 7a shows that the F_1 is smaller when k is smaller (or larger) than the number of real communities because a smaller (or larger) k value leads to inaccurate clustering results. As k approaches the number of real communities, the clustering result gradually approaches the correct result, since the F_1 and Q_l of SRRW are also gradually increasing.

The F_1 w.r.t α and β are shown in Fig. 8a, b respectively. As the value of α increases, F_1 increases rapidly. This is because as α becomes larger, α can encourage further exploration. When α reaches an optimal value, the F_1 begins to

Table 8 Results of effectiveness experiments on five different datasets

Dataset	Metric	RTLCD	TSB	PRN	ACQ	VAC	SRRW-S	SRRW-L	SRRW
Info.		S	S	S	A&S	A&S	A&S	A&S	A&S
	F_1	0.540	0.633	0.471	0.804	0.810	0.841	0.821	0.940
CORA	Q_l	0.618	0.734	0.522	0.957	0.682	0.829	0.741	0.951
	NCR	0.640	0.600	0.320	0.740	0.700	0.910	0.910	0.970
	F_1	0.581	0.624	0.443	0.786	0.784	0.830	0.833	0.891
SINNET	Q_l	0.598	0.711	0.493	0.899	0.711	0.814	0.744	0.873
	NCR	0.650	0.630	0.410	0.660	0.750	0.930	0.920	0.940
	F_1	0.721	0.716	0.397	0.777	0.700	0.871	0.873	0.895
IMDB	Q_l	0.601	0.704	0.477	0.901	0.668	0.831	0.709	0.841
	NCR	0.620	0.600	0.380	0.550	0.540	0.880	0.870	0.980
	F_1	0.715	0.710	0.410	0.759	0.755	0.800	0.813	0.907
DBLP	Q_l	0.497	0.601	0.367	0.907	0.604	0.791	0.681	0.827
	NCR	0.590	0.570	0.350	0.540	0.500	0.870	0.880	0.930
	F_1	0.697	0.700	0.387	0.770	0.790	0.812	0.781	0.921
LFR-2	Q_l	0.634	0.748	0.520	0.961	0.721	0.881	0.876	0.882
	NCR	0.630	0.590	0.400	0.670	0.690	0.900	0.890	0.930
	F_1	0.727	0.693	0.433	0.804	0.781	0.841	0.830	0.881
LFR-5	Q_l	0.705	0.722	0.514	0.948	0.754	0.863	0.820	0.883
	NCR	0.730	0.740	0.430	0.590	0.670	0.860	0.820	0.920

Table 9 Results of effectiveness experiments on three different datasets

Dataset	Metric	SRRW	SRRW-BG	SRRW-NSC
Info.		A&S	A&S	A&S
	F_1	0.940	0.921	0.471
CORA	Q_l	0.951	0.947	0.631
	NCR	0.970	0.930	0.230
	F_1	0.891	0.843	0.855
SINNET	Q_l	0.873	0.869	0.597
	NCR	0.940	0.900	0.240
	F_1	0.895	0.789	0.834
IMDB	Q_l	0.841	0.807	0.613
	NCR	0.980	0.850	0.160

drop slightly. Because the large α impairs the locality property of the restart strategy. For parameter β, SRRW achieves the best result when $\beta = 0.5$. This is because the large β does not make full use of attribute information to assist random walk and the small β ignores the importance of topological information, which causes only nodes that are highly similar to the query node attributes to be captured.

6.5 Effectiveness evaluation (RQ2)

In this section, we focus on SRRW and use real-world and synthetic datasets to evaluate its effectiveness. The specific experimental results are shown in Table 8.

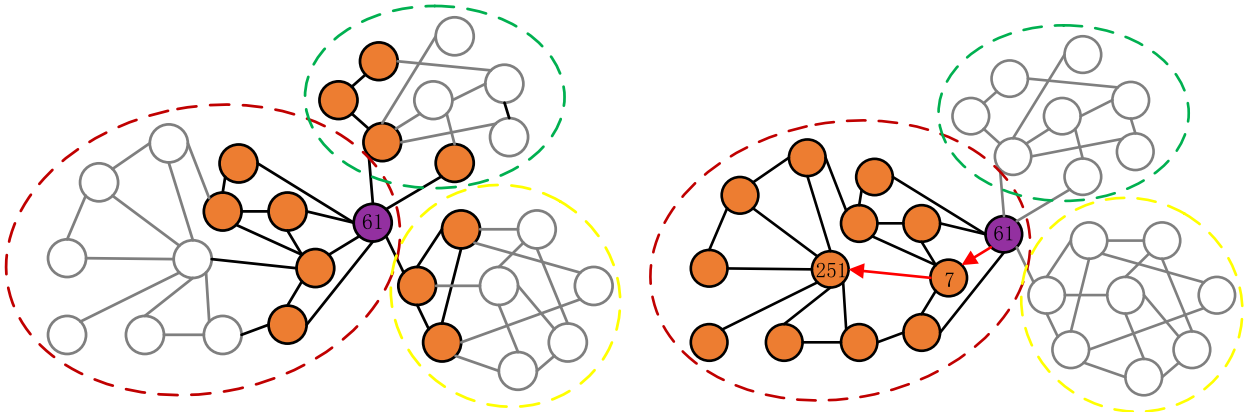
It can be seen from the experimental results that SRRW usually achieves better performance than SRRW-S and SRRW-L. This is because the sigmoid function can not effectively distinguish nodes whose degree exceeds 4. LCC ignores the degree of the node itself.

Table 8 shows that the overall performance of the method that does not use attributes as auxiliary information is lower than other methods. Even if the seed replacement (RTLCD) or core community extension method(TSB), its performance improvement is extremely limited.

From Table 8, we see that, in general, ACQ, VAC, SRRW significantly outperforms all other competitive models, in terms of F_1, NCR, Q_l. It demonstrates the advance of applying attribute information for local attributed community detection. It is worth noting that in all the experimental results, ACQ has achieved the best performance of Q_l. This is attributed to the effectiveness of k-core. SRRW achieved the best results in both F_1 and NCR, which shows that SRRW can effectively avoid the seed dependency problem for any given query node.

6.6 Component contribution analysis (RQ3)

In this section, we consider nodes whose degree is lower than the average degree of the network as low-quality nodes. We randomly select 100 low-quality nodes as query nodes on each real dataset and report the average values of F_1, Q_l, and NCR in Table 9.



(a) SRRW-NSC **(b)** SRRW

Fig. 9 Performance w.r.t. different k on CORA and LFR-2

Similar performance trends are observed for the synthetic datasets. Clearly, our SRRW model significantly outperforms all other competitive models as SRRW-BG and SRRW-NSC. Due to the seed-dependent problem, the performance of SRRW-BG and SRRW-NSC decrease significantly. In summary, for real-world datasets, SRRW has better performance in identifying more ground-truth community members and is more robust to the seed-dependent problem than other algorithms.

To explore how the seed replacement component avoids the seed-dependent problem, Fig. 9 reports the boundary part of the experimental results of SRRW and SRRW-NSC on CORA. Three colored dashed circles respectively identify different real communities. The subgraph composed of orange nodes represents the community located by the corresponding method. From the results in Fig. 9, the experimental results of SRRW-NSC contain many noise nodes. However, in SRRW, the seed-replacement component replaced boundary node 61 with core node 251, so SRRW locates an accurate community.

7 Conclusion

In this paper, to solve the seed-dependent problem, we propose a two-stage community search method based on seed replacement and joint random walk. First, we preprocess the attributed graph via the overlapping clustering method and construct an augmented graph. And then we perform a joint random walk on augmented and use parallel conductance value for community search. Results of comprehensive experiments on bothC and real-world attributed networks verify the advances and effectiveness of SRRW. Although joint random walk can assist community search with the help of attribute information, its essence is to strengthen nodes with similar attributes through the transfer mechanism of node-attribute-node. Our model does not use interactive information between attributes. In the future, we plan to use this interactive information to strengthen random walk and locate attributed subgraphs related to user's preferences.

Acknowledgements This work is supported by the National Natural Science Foundation of China (61762078, 61363058), Gansu Natural Science Foundation Project (21JR7RA114), Research Fund of Guangxi Key Lab of Multi-source Information Mining and Security (MIMS18-08), Northwest Normal University Young Teachers Research Capacity Promotion Plan (NWNU-LKQN2019-2) and Research Fund of Guangxi Key Laboratory of Trusted Software (kx202003).

Data availability statement Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

Akbas E, Zhao P (2017) Truss-based community search: a truss-equivalence based indexing approach. Proc VLDB Endow 10(11):1298–1309

Andersen R, Chung F, Lang K (2006) Local graph partitioning using pagerank vectors. In: 2006 47th annual IEEE symposium on foundations of computer science (FOCS'06). IEEE, pp 475–486

Bian Y, Ni J, Cheng W, Zhang X (2017) Many heads are better than one: local community detection by the multi-walker chain. In: 2017 IEEE international conference on data mining (ICDM). IEEE, pp 21–30

Bian Y, Huan J, Dou D, Zhang X (2020) Rethinking local community detection: Query nodes replacement. In: 2020 IEEE international conference on data mining (ICDM). IEEE, pp 930–935

Chang Y, Ma H, Chang L, Li Z (2022) Community detection with attributed random walk via seed replacement. Front Comput Sci 16(5):1–12

Chen L, Liu C, Zhou R, Li J, Yang X, Wang B (2018) Maximum co-located community search in large scale social networks. Proc VLDB Endow 11(10):1233–1246

Cheng J, Ke Y, Fu AWC, Yu JX, Zhu L (2011) Finding maximal cliques in massive networks. ACM Trans Database Syst (TODS) 36(4):1–34

Cheng J, Ke Y, Chu S, Özsu MT (2011b) Efficient core decomposition in massive networks. In: 2011 IEEE 27th international conference on data engineering. IEEE, pp 51–62

Cui W, Xiao Y, Wang H, Lu Y, Wang W (2013) Online search of overlapping communities. In: Proceedings of the 2013 ACM SIGMOD international conference on Management of data, pp 277–288

Cui W, Xiao Y, Wang H, Wang W (2014) Local search of communities in large graphs. In: Proceedings of the 2014 ACM SIGMOD international conference on management of data, pp 991–1002

Ding X, Zhang J, Yang J (2018) A robust two-stage algorithm for local community detection. Knowl Based Syst 152:188–199

Fang Y, Cheng R, Luo S, Hu J (2016) Effective community search for large attributed graphs. Proc VLDB Endow 9(12):1233–1244

Fang Y, Cheng R, Chen Y, Luo S, Hu J (2017) Effective and efficient attributed community search. VLDB J 26(6):803–828

Fang Y, Huang X, Qin L, Zhang Y, Zhang W, Cheng R, Lin X (2020) A survey of community search over big graphs. VLDB J 29(1):353–392

Hsu CC, Lai YA, Chen WH, Feng MH, Lin SD (2017) Unsupervised ranking using graph structures and node attributes. In: Proceedings of the tenth ACM international conference on web search and data mining, pp 771–779

Huang X, Lakshmanan LV (2017) Attribute-driven community search. Proc VLDB Endow 10(9):949–960

Huang X, Cheng H, Qin L, Tian W, Yu JX (2014) Querying k-truss community in large and dynamic graphs. In: Proceedings of the 2014 ACM SIGMOD international conference on management of data, pp 1311–1322

Huang X, Lakshmanan LV, Xu J (2017) Community search over big graphs: models, algorithms, and opportunities. In: 2017 IEEE 33rd international conference on data engineering (ICDE). IEEE, pp 1451–1454

Li J, Ma H, Li Q, Li Z, Chang L (2021) A two-stage community search method based on seed replacement and joint random walk. In: 2021 international joint conference on neural networks (IJCNN). IEEE, pp 1–7

Li Q, Ma H, Li J, Li Z, Jiang Y (2022) Searching target communities with outliers in attributed graph. Knowl Based Syst 235:107622

Liu S, Xia Z (2020) A two-stage BFS local community detection algorithm based on node transfer similarity and local clustering coefficient. Phys A 537:122717

Liu Y, Ma H, Liu H, Yu L (2020) An overlapping subspace k-means clustering algorithm. Comput Eng 46:58–63

Liu Q, Zhu Y, Zhao M, Huang X, Xu J, Gao Y (2020b) VAC: vertex-centric attributed community search. In: 2020 IEEE 36th international conference on data engineering (ICDE). IEEE, pp 937–948

Luo W, Lu N, Ni L, Zhu W, Ding W (2020) Local community detection by the nearest nodes with greater centrality. Inf Sci 517:377–392

Luo J, Cao X, Xie X, Qu Q, Xu Z, Jensen CS (2020b) Efficient attribute-constrained co-located community search. In: 2020 IEEE 36th international conference on data engineering (ICDE). IEEE, pp 1201–1212

Nascimento MC (2014) Community detection in networks via a spectral heuristic based on the clustering coefficient. Discrete Appl Math 176:89–99

Sozio M, Gionis A (2010) The community-search problem and how to plan a successful cocktail party. In: Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining, pp 939–948

Tong H, Faloutsos C, Pan JY (2006) Fast random walk with restart and its applications. In: 6th international conference on data mining (ICDM'06). IEEE, pp 613–622

Wang J, Cheng J (2012) Truss decomposition in massive networks. arxiv:1205.6693

Yin H, Benson AR, Leskovec J, Gleich DF (2017) Local higher-order graph clustering. In: Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining, pp 555–564

Yuan L, Qin L, Zhang W, Chang L, Yang J (2017) Index-based densest clique percolation community search in networks. IEEE Trans Knowl Data Eng 30(5):922–935

Zhao Q, Ma H, Li X, Li Z (2021) Is the simple assignment enough? Exploring the interpretability for community detection. Int J Mach Learn Cybern 12(12):3463–3474

Zhao Q, Ma H, Guo L, Li Z (2022) Hierarchical attention network for attributed community detection of joint representation. Neural Comput Appl 34(7):5587–601

Zhe C, Sun A, Xiao X (2019) Community detection on large complex attribute network. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining, pp 2041–2049