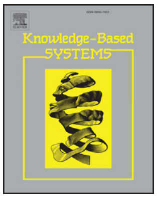
Knowledge-Based Systems 150 (2018) 85–94

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys



Dissimilarity-constrained node attribute coverage diversification for novelty-enhanced top-*k* search in large attributed networks

Zaiqiao Meng[a], Hong Shen[a,b,*]

[a] School of Information Science and Technology, Sun Yat-Sen University, China
[b] School of Computer Science, University of Adelaide, Australia

ARTICLE INFO

Article history:
Received 19 March 2017
Revised 28 February 2018
Accepted 2 March 2018
Available online 17 March 2018

Keywords:
Information retrieval
Query diversification
Top-k search
Submodular maximization

ABSTRACT

Query diversification as an effective way to reduce query ambiguity and enhance result novelty has received much attention in top-*k* search applications on large networks. A major drawback of the existing diversification models is that they do not consider redundancy elimination during the course of search, resulting in unassured novelty in the search result. In this paper, to improve the novelty of the search result, we propose a new method of diversified top-*k* similarity search by combining diversification of node attribute coverage with a dissimilarity constraint. Due to the non-monotonicity implied by the dissimilarity constraint, existing techniques based on monotonicity assumptions cannot be applied. Our model requires solving a new problem of *Dissimilarity Constrained Non-monotone Submodular Maximization* (DCNSM). Based on constructing a dissimilarity-based graph, we solve this problem by a greedy algorithm achieving an approximation ratio of $1/\Delta$, where Δ is the maximum node degree of the dissimilarity-based graph, in time linear to the number of edges of the graph. We show that DCNSM cannot be approximated in ratio $|V|^{1-\epsilon}$, indicating that our solution achieves an optimal ratio. We conduct extensive experiments on both synthetic and real-world attributed network datasets. The results show that our diversification model significantly outperforms the baseline methods, and confirm that combining dissimilarity constraint in diversification can significantly improve the novelty of search result.

1. Introduction

Searching for top-*k* nodes similar to a given query request in a network has numerous applications including graph clustering [15], graph query [27], object retrieval and recommendation [34]. There has been substantial research on ranking nodes and their similarity (proximity) estimation, such as the Personalized PageRank [14] and SimRank [18]. These basic methods and their variations such as P-Rank [38], TopSim [23] and Panther [37] have been successfully applied in a wide range of applications.

Nowadays with rich information available from online social networks, real social entities and their relationships can be built in a network in which nodes are associated with a set of attributes describing their properties and edges representing relationships among the nodes. In such circumstances, the problem of searching for similar nodes to a given node becomes more sophisticated and challenging.

Firstly, the top-*k* nodes resulted from the traditional similarity search methods are often highly related. It is hard to get desired similar results with such few and highly related nodes. Search result diversification has been widely studied as a way of tackling query ambiguity and enhancing result novelty in information retrieval [6,12]. Most of these diversified models tried to trade off relevance and diversity according to some parameter. In the literature there are many studies on modeling the search result diversification for network datasets. Expansion ratio [24] and expanded relevance [20] are two representative diversification models proposed recently based on node's ego features (e.g. neighborhood) diversification and addressed by the solution of the classic Monotone Submodular Maximization problem. However, a major drawback of these models is that they give no explicit mechanism for redundancy elimination, resulting in lack of novelty in their search results.

Moreover, a network with attributes has more complex characters. More precisely, node attributes with the links among them provide rich and complementary sources of information that should be used for revealing, understanding and exploiting the latent diversified structure in attributed network data. For example, in social networks users have profile information, in

[*] Corresponding author.
E-mail addresses: zqmeng@aliyun.com (Z. Meng), hongsh01@gmail.com (H. Shen).

https://doi.org/10.1016/j.knosys.2018.03.008
0950-7051/© 2018 Elsevier B.V. All rights reserved.

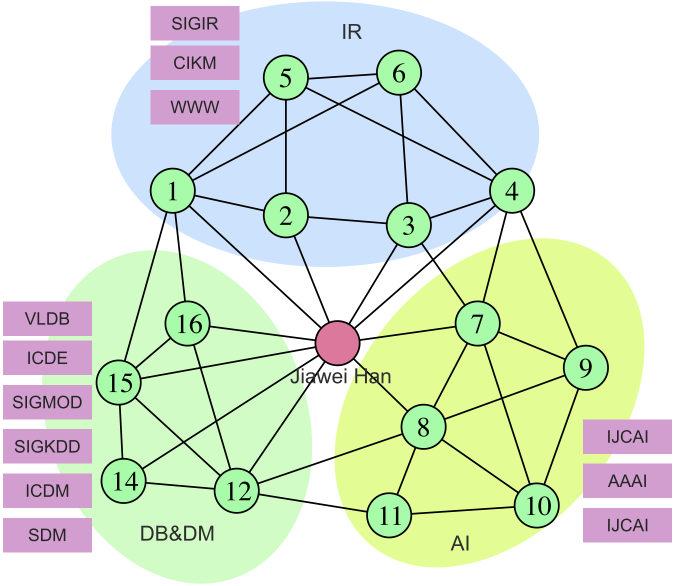


Fig. 1. A portion of Jiawei Han' s academic ego social network. 1

document networks each node also contains the text of the document that it represents. Fig. 1 depicts a portion of Jiawei Han's academic ego social network extracted from DBLP.[1] Three research communities, IR, DB&DM and AI, are distinguished by three cluster based on their common attributes (the conferences they have published papers), which are the non-topological features. Therefore, diversification should also be in presence for these non-topological features. However, the existing diversification models focus purely on network topologies only, and totally ignore that attributes also expose diversity that should be fully exploited to meet the ambiguous query intend.

In this paper, to model the diversification search problem in attributed networks, we first formulate a problem that considers only the *attribute coverage diversification* (ACD). We show that the optimization objective of the ACD problem is a non-decreasing submodular function, and give a marginal gain based greedy algorithm that yields a $(1-1/e)$-approximation near-optimal solution. To further improve novelty of the search result, we extend the ACD problem with the r-dissimilar constraint that captures the dissimilarity between result nodes based on the topological structure of the graph. We show that the new problem becomes more complicated, because adding a new node to the result set following the monotonicity may break the dissimilarity constraint and hence the existing techniques cannot be applied any more. Our model requires to solve a new problem called *Dissimilarity Constrained Non-monotone Submodular Maximization* (DCNSM). Based on constructing a dissimilarity-based graph, we propose a greedy algorithm achieving an approximation ratio of $1/\Delta$, where Δ is the maximum degree of its dissimilarity-based graph, and runs in $O(k(\overline{a}|V|+|E|))$ time.

The main contributions of this paper are: (1) We formulate the problem of node*attribute coverage diversification* (ACD) for top-k similarity search in attributed networks as that of maximizing a monotone submodular function, and give a $(1-1/e)$-approximation near-optimal solution. (2) We formulate the dissimilarity-constrained node attribute coverage diversification problem, an extended ACD problem, as that of maximizing a dissimilarity-constrained non-monotone submodular function. We prove that no $|V|^{1-\epsilon}$-ratio approximation scheme exists for this problem for any $\epsilon > 0$ and present a linear-time (to $|E|$) algorithm achieving the optimal approximation ratio $1/\Delta$, where Δ is the maximum degree of its dissimilarity-based graph. (3) We conduct extensive experiments on both synthetic network datasets and real-world attributed network datasets, and the results show that our proposed dissimilarity-constrained node attribute coverage diversification method significantly outperforms other methods.

2. Related work

The work presented in this paper is closely related to *similarity search on networks, submodular function maximization*, and *search result diversification on networks.*

2.1. Similarity search on networks

There have been various measures to estimate similarity between nodes on networks. Personalized PageRank (PPR) [14] is a random walk based measure evolved from the classic PageRank algorithm [29]. Similar to PPR, SimRank is defined recursively with respect to the "random surfer-pairs model", and it evaluates the similarity between two nodes as the first-meeting probability of two random surfers. Existing algorithms like P-Rank [38], TopSim [23] are extensions of SimRank. Some other examples include discounted/truncated hitting time [31], penalized hitting probability [36], and nearest neighbor [2,35] are also referred to as the random walk based method. Recently, a random path sampling based method—Panther [37] was proposed that can provably estimate the similarity between nodes efficiently and accurately.

2.2. Submodular function maximization

Submodularity is a property of set functions with deep theoretical consequences and far-reaching applications. Submodular set functions have been widely applied to many fields, including document summarization [25], image segmentation [17], sensor placement [19], diversifying search [3,24], and algorithmic game theory [8]. Submodular function maximization captures classic NP-hard problems in the combinational optimization such as *max cut, maximum facility location* and *max k-cover* problems [4,5,11,33]. Some studies dealt with submodular function maximization subject to various combinatorial constraints, such as the bases of a matroid [33], multiple knapsack constraints [22] and submodular knapsack [16]. To the best of our knowledge, there is no published work for the problem of maximizing submodular functions subject to a dissimilarity (distance) constraint.

2.3. Search result diversification on networks

There are several studies on search results diversification in network data. DivRank [26] employs a time-variant random walk process to facilitates the rich-gets-richer mechanism in node ranking. Tong, et al. [32] propose a scalable diversified ranking algorithm by optimizing a predefined diversified goodness measure. Recently a neighbor expansion based diversified ranking method was proposed, with the assumption that nodes with large expansion would be dissimilar to each other [24]. Küçüktunç [20] proposed a measure called expanded relevance which combines both relevance and diversity into a single function in order to measure the coverage of the relevant part of the graph. These methods are designed to work with the simplified structural network without considering node attributes. Our diversification model focuses on the attributed networks, and combines with a dissimilarity constraint as an explicit measure to eliminate redundancy.

[1] http://dblp.uni-trier.de/xml/.

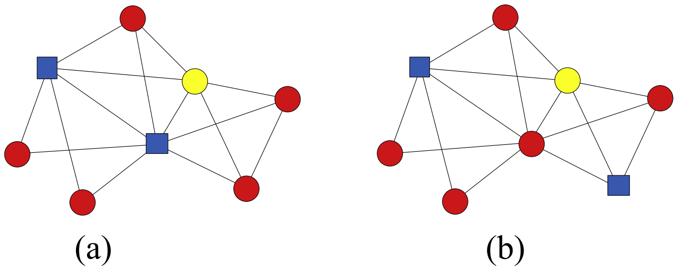


Fig. 2. (a) Diversified result from EP1 (b) A novelty-enhanced diversified result.

3. Problem formulation

3.1. Preliminaries

Let $G = (V, E, W_E)$ be an undirected weighted graph with $|V|$ nodes and $|E|$ edges, where $W_E = \{w:\ E(G) \to R+\}$ is the set of edge weights. If W_E is not specified, all edges have weight 1. Let $V(G)$ be the set of nodes in G, $N_G(u)$ be the neighborhood of node u, and $N_G^+(u) = \{u\} \bigcup N_G(u)$. We denote by $d_G(u)$ the degree of node $u \in V$, and by Δ the maximum degree of G.

Given a G and a query node $q \in V$, we use similarity score $s(u)$ to measure the relevance of node u to q, and dissimilarity score $diss(u, v)$ to measure the dissimilarity between nodes u and v. They are computed purely based on the topological structure of the nodes.

Given an element u, a ground set N, a function $f: 2^N \to R^+$ is called *submodular function* if for $S \subseteq T \subseteq N$ and $u \in N \backslash B$, $f_u(S) \geq f_u(T)$, which $f_u(S) = f(S+u) - f(S)$ is called the *marginal gain*. A submodular function f is *monotone* if for every $S \subseteq T$ we have that $f(S) \leq f(T)$ [28].

3.2. Problem definition

We address the problem of diversifying top-k similarity search results (nodes) for a given node q in graph G, to meet the user's ambiguous query intent. This problem has many interesting applications in the areas such as social-network recommendation, collaborative event organization and tag suggestion. For instance, in a photo sharing social network such as *Pinterest*, given a new photo being uploaded with initial tags that the user provides for this photo, the system may suggest to the user a number of additional tags. A good suggestion is a set of diverse tags related to the user's friends, so that the user can quickly choose the proper tags for the new photo.

We assume that the relevance score of each node w.r.t q has already been obtained by a given similarity search algorithm. The main challenge of this problem is how to properly measure the diversification. Some previous models try to optimize a function with a single diversification measure based on the ego features of nodes, such as neighborhood. For example, Li and Yu [24] formulates this problem as a bicriteria objective optimization problem that trades off the relevance and the*expansion ratio*, while Küçüktunç et al. [20] optimizes a single function (called *expanded relevance*) which combines both relevance and neighborhood diversity. These neighbor expansion based methods are based on the intuition that nodes with a large neighborhood are dissimilar to each other, thus leading to high diversity. This is, however, not always true, as illustrated in Example 1.

Example 1. Consider a graph in Fig. 2 with a yellow query node, and suppose that other nodes has a same relevance score to the query node. The blue square nodes in Fig. 2(a) is the possible result returned by EP1 algorithm [24] and the circle nodes denote their expansion nodes. It is clear that result nodes in Fig. 2(b) are more novel to each other than Fig. 2(a), though the expansion ratio of the selected nodes in both figures is 1. Because the selected nodes in Fig. 2(b) are more dissimilar to each other whether it is measured based on shortest path or based on common neighbors.

This is a common problem in popular diversified search and recommendation algorithms [7,20,21,24,26] that pollute the top-k result or recommendations with many similar nodes, i.e., redundancy. It is typically not interesting to be recommended similar users if the query user have a wide interest. Therefore, a good diversification metric should choose a set of representative nodes that are dissimilar to each other (i.e., eliminate redundancy).

Moreover, in real-world social networks with rich categorical node attributes, diversified search should consider both network structure and node attribute information. Ideally, the representative nodes should cover categorical node attribute information as much as possible so that the result have more chance to meet the query intent of average users. Our main task of this paper is to address diversified search in attributed networks that considers both network structure and node attribute information with a desired trade-off between structure dissimilarity and attribute diversity.

An attributed network is an undirected weighted graph augmented by node attributes: $G_A = (G, A_V) = (V, E, W_E, A_V)$, where $A_V = \left\{A' : V(G) \to S_A\right\}$ is the *node* attribute-weighting function, S_A is a family of subsets of attribute set A_V.

We assume that each attribute is binary-valued. Other types of attribute variables can be discretized into categorical variables. For example, in Facebook social network, various universities (e.g., MIT, CMU, and Stanford) in user profile are directly treated as separate binary attributes, the posted status updates can be extracted as the binary keyword attributes by using stemming, tokenization, and stopword removal etc. techniques.

To model diversification in attributed networks, we first introduce the *attribute coverage diversification* problem which only considers the diversification on attribute features of nodes, then we propose a new model of diversification combining diversification of node attribute coverage with a node dissimilarity constraint such that any two nodes in the search result must satisfy a given dissimilarity threshold to ensure the novelty, and cover attributes as many as possible.

Definition 2. Attribute Coverage. The *attribute coverage* of node v, $|A_v|$, is the cardinality of attribute set of v, and $|A_S| = |\cup A_v|$ is the *attribute coverage* of node subset S. The *attribute coverage ratio* (ACR) of S, $ACR = \frac{|A_S|}{|A|}$, is defined as the normalized attribute coverage representing the coverage diversity, where A_V is the base attribute set (universe).

Problem 3. Attribute CoverAGE Diversification (ACD). Given an undirected weighted attributed network $G_A = (V, E, W_E, A_V)$, a query node q, a relevance metric s, and a positive integer k, the problem is to find a subset $S \subseteq V$ such that:

$$\max_{S\subseteq V} f(S) = \max_{S\subseteq V}\left\{(1-\lambda)\sum_{u\in S} s(u) + \lambda\frac{|A_S|}{|A|}\right\}$$
$$s.t.\quad |S| = k. \tag{1}$$

where $\lambda \in [0, 1]$ is a parameter to trade off relevance and diversity.

Problem 3 utilizes the linear combination of the attribute coverage and relevance metric as the optimization objective, and aims at finding k nodes such that the combination of the relevance to q and the attribute coverage of the result node set S achieves the maximum. The definition of attribute diversity based on attribute coverage is intuitive and reasonable. It indicates that the more attributes covered by nodes, the more diverse the result set is. However, the ACD problem only considers one diversification measure that is the attribute diversification. To improve the novelty of result, we further take into account of node dissimilarity as their topological structure diversification. The new problem is defined as follows.

Problem 4. *r*-Dissimilar Attribute Coverage Diversification (*r*-DACD). Given an undirected weighted attributed network $G = (V, E, W_E, A_V)$, a query node *q*, a relevance metric *s*, a dissimilarity metric *diss*, a dissimilarity threshold *r*, and a positive integer *k*, the problem is to find a subset $S \subseteq V$ such that:

$$\max_{S\subseteq V} f(S) = \max_{S\subseteq V}\left\{(1-\lambda)\sum_{u\in S} s(u) + \lambda\frac{|A_S|}{|A|}\right\}$$

$$s.t. \quad |S| = k, \qquad (2)$$
$$\forall v_1, v_2 \in S, \quad diss(v_1, v_2) \geq r.$$

In this problem, the *r*-dissimilar constraint scatters the result to a wider range of topological structure space, describing the topological structural diversification of the result set. The relevance measurement and dissimilar metrics in Problems 3 and 4 will be discussed in the next section.

3.3. Connection to neighbor expansion based methods

There exist many previous studies [24,26] on bicriteria optimization measures for diversified graph search. For example, Li et al. [24] tried to optimize the following diversified bicriteria optimization objective:

$$\max_{S\subseteq V} f(S) = \max_{S\subseteq V}\left\{(1-\lambda)\sum_{u\in S} s(u) + \lambda\frac{|N_S|}{|N|}\right\}. \qquad (3)$$

The first term is the sum of the Personalized PageRank scores over the results, which reflects the relevance. The second term is the expansion ratio of the results, which reflects the diversity.

We notice that our ACD problem is a generalization of the neighbor expansion diversification. If we regards the neighbor sets of nodes as their attribute sets, the neighbor expansion based diversification will become our ACD problem. In other words, neighbors are taken as a certain type of node attributes.

Another neighbor expansion based measure is called *expanded relevance* that combines both relevance and diversity into a single function in order to measure the coverage of the relevance part of the graph [20]. They argued that bicriteria optimization is inappropriate, because the both measures of these diversification methods seem highly correlated between each other. It is worth mentioning that both ACD and *r*-DACD also optimize a bicriteria objective, the structure similarity and the attribute diversity, but they stand as two independent metrics to measure relevance and diversity respectively. To capture the novelty among result nodes, we also add a node dissimilarity constraint to these neighbor expansion based methods. We will illustrate the comparison of these dissimilarity constrained models in detail in the follow-up experiments.

4. The algorithms

4.1. Relevance metric and dissimilarity metric

Given a large graph, finding top-*k* most similar nodes to a given query node is a fundamental problem. Most previous work about diversified search on graphs utilizes Personalized PageRank (PPR) [14] as their relevance metric. In this paper we use Panther [37] to measure the relevance to query node and the dissimilarity of node pairs in the result set, for the following reasons. Firstly, compared with PPR, Panther has a lower time complexity than PPR. Secondly, we need a unified measure to accommodate both relevance and dissimilarity of nodes, and using Panther can easily achieve this goal.

The basic idea of Panther is that two nodes are similar if they frequently appear on the same paths. The algorithm randomly selects a vertex in *G* as the starting point, and then conducts random walks of *T* steps from *v* using the weight proportion as the transition probability to corresponding nodes. The relevance score is defined as: $s(u) = \frac{|p_{u,q}|}{R}$, where $|p_{u,\,q}|$ is the number of paths that contains node *u* and the query node *q*, and *R* is the total number of random paths. The authors showed that Panther can accurately and quickly estimate the similarity between all-pairs of nodes [37].

In this paper, we evaluate the dissimilarity of nodes using the same idea as Panther, and calculate the normalized dissimilarity score by $diss(u, v) = 1 - \frac{|p_{u,v}|}{|p_{max}|-|p_{min}|}$, where $|p_{u,\,v}|$ is the number of random paths between node *u* and node *v*, $|p_{max}|$ and $|p_{min}|$ are respectively the maximum and minimum of this number among all pairs of nodes.

4.2. The greedy ACD algorithm

It is easy to see that the ACD problem is NP-hard, because if we set $\lambda = 1$, the problem is equivalent to the *max k-cover* problem which is known to be NP-hard [10]. Thus we resort to develop approximate algorithms for this problem. Below, we show that Problem 3 is a nondecreasing submodular function with a cardinality constraint [4].

Theorem 5. *The optimization problem of f(S) defined in Problem 3 is a Cardinality Constrained Monotone Submodular Maximization problem.*

Theorem 5 can be proved in the same way as that of Theorem 3.2 in [24].

Although Problem 3 is NP-hard, there exist efficient approximation algorithms for solving the submodular function maximization problem [28]. Here, we show a greedy solution based on [24] that repeatedly chooses a node with the *maximum marginal gain* is a near-optimal approximation solution for the problem. Algorithm 1 outlines the *greedy ACD* algorithm (GACD).

Algorithm 1 Greedy ACD.

```
Input: $G$, $k$, $\lambda$, query node $q$, relevance metrics$(\cdot)$ to $q$
Output: A set $S$ with $k$ diversified nodes
1: $S \leftarrow \emptyset$
2: while $|S| < k$ do
3:     update $f_u(S)$ for nodes in $G$;
4:     $v \leftarrow \max_{u\in V} f_u(S)$;
5:     $S = S \bigcup \{u\}$;
6:     $V = V \backslash u$; $E = E \backslash \{(v_1 \in V,\ v_2 \in u\}$
7: return $S$;
```

In this algorithm, most of the time cost is for computing the marginal gain $f_u(S)$ of every node (Step 3). We can use an indicator array to represent the attribute coverage, then the time complexity for Steps 3 and 4 is $O(\overline{a}|V|)$, where $\overline{a}$ represents the average number of attributes belong to a node. In the real large social networks, though the total attribute size $|A|$ might be large, the average number of available attribute of a single user $\overline{a}$ is far small than the user size $|V|$ and can be viewed as a small constant. Thus, the total time complexity is $O(k\overline{a}|V|)$, which is linear in the number of nodes.

Theorem 6. *Algorithm 1 is a* $(1 - 1/e)$*-approximation algorithm for Problem 3.*

The correctness of the theorem can been seen from the result in [28]: For a monotone submodular function, greedy construction by selecting an element with the maximum marginal gain yields an $(1 - 1/e)$-approximation to the optimal solution.

4.3. The greedy r-DACD algorithm

The main difficulty for solving the r-DACD problem (Problem 4) is that incorporating the dissimilarity constraint makes the optimization objective become non-monotone. This is because adding a new node to the result set may break the dissimilarity constraint. Therefore, the greedy algorithm used to solve the *monotone submodular maximization* problem cannot be deployed to solve the r-DACD problem. As far as we know, there is no reported result on solving this dissimilarity constrained problem. We denote this problem as *Dissimilarity Constrained Non-monotone Submodular Maximization* (DCNSM) problem: maximizing $f(S)$ such that $\forall v_1, v_2 \in S$, $diss(v_1, v_2) \geq r$.

In fact, the following theorem shows that the DCNSM problem is NP-hard.

Theorem 7. *The DCNSM problem is no easier than the k-Clique problem.*

Proof. We prove it by reducing the k-Clique problem, a well known NP-hard problem, to DCNSM.

For an arbitrary graph $G(V, E)$, we consider the k-Clique problem of finding a complete subgraph spanning k nodes in G. We reduce this problem to the following instance of DCNSM: $\max\limits_{S\subseteq V} f(S) = \max\limits_{S\subseteq V} \sum_{u,v\in S} diss(u, v)$ s.t. $|S| = k$, $\forall u, v \in S$: $diss(u, v) \geq 1$, where $diss(u, v) = 1$ if $(u, v) \subseteq E$ and 0 otherwise, for all $u, v \in V$. It is obvious that $f(S)$ is submodular set function. If we find a solution S^* with a maximum of $f(S)$ (i.e. $f(S) = k(k-1)/2$) to that instance of DCNSM then S^* is a node set of a k-Clique in G. Thus the theorem holds. □

Following from this theorem, we can even draw the conclusion that it is hard to find an approximate within ratio $|V|^{1-\epsilon}$ for DCNSM, since it is known [13] that the Max-Clique problem, i.e., the k-Clique for the maximum k, cannot be approximated within $|V|^{1-\epsilon}$ in polynomial-time, for any $\epsilon > 0$, unless $NP = ZPP$.

Given an arbitrary DCNSM problem, we can generate a *dissimilarity-based graph* G_d according to its dissimilarity metric, i.e., for $\forall v_1, v_2 \in S$, $diss(v_1, v_2) \geq r$, we create an edge (v_1, v_2) in G_d, otherwise, $(v_1, v_2) \notin E$. Then the DCNSM problem is a maximum weighted independent set problem on G_d, where the weight is computed by a submodular set function. Inspired by Sakai et al. [30], we present the following greedy algorithm (Algorithm 2) for solving the DCNSM problem.

Algorithm 2 Greedy r-DACD.

```
Input: $G$, $k$, $\lambda$, query node $q$, relevance metrics$(\cdot)$ to $q$, a dissimilarity metric $diss(\cdot, \cdot)$
Output: A set $S$ with $k$ diversified nodes
1: generate dissimilarity-based graph $G_d$;
2: $S \leftarrow \emptyset$;
3:  while $|S| < k$ do
4:    update $f_u(S)$ for nodes in $G_d$;
5:    $v \leftarrow \max_{u\in V} f_u(S)$
      $s.t. f_u(S) \geq \frac{1}{\Delta} \sum_{w\in N_{G_d}(u)} f_w(S)$;
6:    $S = S \bigcup \{u\}$;
7:    $V = V \backslash N^+_{G_d}(u)$; $E = E \backslash \{(v_1 \in V,\ v_2 \in N^+_{G_d}(u)\}$
8: return $S$;
```

Algorithm 2 first constructs a dissimilarity-based graph G_d, then iteratively selects a node with the maximum marginal gain that is greater than $1/\Delta$ of the summed marginal gain of its neighbors. In each iteration, the algorithm computes the marginal gain $f_u(S)$ for each $u \in V(G_d)$, and chooses a node u with maximum $f_u(S)$ that satisfies the condition: $f_u(S) \geq \frac{1}{\Delta} \sum_{w\in N_{G_d}(u)} f_w(S)$, which we call as the*local maximal node*. The following lemma shows that Algorithm 2 can always find such a *local maximal node* in each iteration.

Lemma 8. *There always exists a local maximal node in G_d.*

Proof. Assume there is no local maximal node in G_d, then $f_u(S) < \frac{1}{\Delta} \sum_{w\in N_{G_d}(u)} f_w(S)$ holds for $\forall u \in G_d$

Accumulating this inequality for each nodes in G_d, we can get:

$$\sum_{u\in G_d} f_u(S) < \sum_{u\in G_d} \frac{1}{\Delta} \sum_{w\in N_{G_d}(u)} f_w(S)$$

Collecting $f_w(S)$ of right side according w's neighbors and Rearranging yields:

$$\sum_{u\in G_d} \frac{\Delta}{\Delta} f_u(S) < \sum_{u\in G_d} \frac{d(u)}{\Delta} f_u(S)$$

From $d(u) \leq \Delta$, we reach a contradiction, hence the lemma holds. □

So the algorithm terminates after k iterations.

Because in each iteration the algorithm finds a node u whose $f(S)$ is at least $\frac{1}{\Delta}$ of that node with maximum $f(S)$, we have the following theorem:

Theorem 9. *Algorithm 2 gives a $\frac{1}{\Delta}$-approximation solution for Problem 4, where Δ is the maximum degree of its dissimilarity-based graph.*

Proof. Let $OPT(G_d)$ be optimal value for G_d, and $v_i \in S$ be a sequence of nodes of solution by Algorithm 2. The algorithm starts with $S = \{\emptyset\}$, and ends when $|S| = k$ with an optimal value of $\sum_{i=1}^{k} f_{v_i}(S)$. Because we greedily select the maximum $f_{v_i}(S)$ of the local maximal node at each iterator, w.l.o.g., we consider the maximum k in G_d. That is, after k iterators of Algorithm 2 all the nodes are removed.

Now we consider the induced subgraph $G_d(N^+_{G_d}(v_i))$. The optimal solution of $G_d(N^+_{G_d}(v_i))$ might be v_i or a subset of $N_{G_d}(v_i)$, and the optimal value satisfies:

$$OPT\left(G_d\left(N^+_{G_d}(v_i)\right)\right) \leq \max\left(f_{v_i}(S), f_{N_{G_d}(v_i)}(S)\right)$$

From the local maximal condition, we have:

$$f_u(S) \geq \frac{1}{\Delta} \sum_{w\in N_{G_d}(u)} f_w(S)$$

Combining the two inequalities we have:

$$f_{v_i}(S) \geq \frac{1}{\Delta} OPT\left(G_d\left(N^+_{G_d}(v_i)\right)\right)$$

Since each induced subgraph $N^+_{G_d}(v_i)$ is disjointed, according to the submodularity of f, the following inequality holds:

$$\sum_{i=1}^{k} OPT\left(G_d\left(N^+_{G_d}(v_i)\right)\right) \geq OPT(G_d)$$

Then we have:

$$\sum_{i=1}^{k} f_{v_i}(S) \geq \sum_{i=1}^{k} \frac{1}{\Delta} OPT\left(G_d\left(N^+_{G_d}(v_i)\right)\right) \geq \frac{OPT(G_d)}{\Delta}$$

Thus the theorem holds. □

Now we analyze the time complexity of Algorithm 2. For each loop, computing the marginal gain $f_u(S)$ for every nodes in G_d (Step

Table 1
Summary of the real-world datasets.

Datasets	Nodes	Edges	Attributes
Facebook	4039	88,234	1406
DBLP	73,242	373,797	172
AMiner	1,560,640	4,258,946	2,868,034

4) needs $O(\overline{a}|V|)$ time. Finding the *local maximal node* in Step 5 takes $O(|E|)$ time. The loop terminates in k iterations. Hence the time complexity of G*r*DACD is $O(k(\overline{a}|V|+|E|))$.

The limitation of Algorithm 2 is that it cannot achieve any constant approximation ratio. However, in the worst case $\Delta = |V|$, G*r*DACD still achieves a tight [13] approximation guarantee according to Theorem 7. In practice, our experimental results on the real-world data show that the algorithm achieves good performance.

5. Experiment setup

5.1. Datasets

For our experiments, we evaluate the effectiveness and efficiency of our proposed algorithms on both synthetic network datasets and real-world network datasets.

We consider three real-world datasets where we have the information of network topology and node attributes summarized in Table 1. The Facebook dataset is downloaded from SNAP.[2] This dataset is built from profile and relation data from 10 users' ego-networks in Facebook, and the attributes are constructed by their user profiles. The DBLP dataset is from the DBLP public bibliography data. We build a coauthor network extracted from the papers in top 172 conferences (rank A and B) from 10 research areas ranked and classified by CCF.[3] We treat authors and their collaborations in papers as nodes and edges respectively. The 172 conferences are viewed as the attributes of each node. The AMiner coauthor dataset is downloaded from AMiner.org.[4] This network dataset is also built by collaboration relationships among the authors. But unlike the DBLP dataset, the attributes in AMiner dataset are constructed by the extracted keyterms of their papers.

We generate a collection of synthetic networks by using two random network models: the Erdös-Rényi (ER) random network model [9] and the Barabási–Albert (BA) scale-free network [1]. We also generate binary-valued attributes randomly for the nodes in these networks.

5.2. Evaluation metrics

In the literature, there are no universal measures for diversified search in graphs. Different measures are applied depending on their different "diversification" definitions. In our experiments, to measure topological structure diversity, we employ the metric of *density* [26] of the induced subgraph of the result set, which is the number of edges in the graph divided by the maximum possible number of edges of the graph. In the topological structure of the network, smaller density implies higher diversity of the result. We also introduce two new metrics for our new problems: the *attributed coverage ratio* (*ACR*) and the *minimum dissimilarity* (*MinDiss*). *ACR* is defined in Definition 2, which measures the attribute diversity. *MisDiss* is defined as $MinDiss = Min_{v_1,v_2 \in S} diss(v_1, v_2)$, which measures the structure diversity.

5.3. Baselines

We compare our proposed methods with several state-of-the-art baselines: the bicriteria expansion optimization [24] (denoted by EP) and the expanded relevance [20] (denoted by BC). For our experiments, we mainly focus on l-step, for $l = 1$ and $l = 2$, denoted by EP1, EP2 and BC1, BC2 respectively. As EP1, EP2, BC1 and BC2 can also be built on dissimilarity constraint, we also consider four ***r***-Dissimilar constraint variants of these methods, denoted by *r*DEP1, *r*DEP2 and *r*DBC1, *r*DBC2 respectively.

5.4. Parameter settings

We conduct experiments by first running the Panther algorithm to obtain the relevance score and dissimilarity score for all pairs of nodes, then running the diversified search algorithms to obtain their diversified search results. There are two parameters in Panther: path length T and error-bound ϵ. We empirically set $T = 5$ and $\epsilon \approx \sqrt{1/|E|}$ as they are known to produce good performance on all the datasets [37]. In the diversified search algorithms, there are two parameters: λ used to trade off relevance and diversity, and r used to restrict the dissimilarity of results. In the bicriteria optimization algorithms (e.g. GACD, G*r*DACD, EP1, EP2, DEP1 and DEP2), the tradeoff parameter λ is set to the value of their best performance respective to comparison experiments on all metrics, as this parameter has different scales among different diversity measures. We empirically set $r = 0.9$ for those dissimilarity constraint algorithms in our experiments.

All experiments are conducted on an Ubuntu 14.04 server with two Intel Xeon E5-2683 v3 (2.0 GHz) CPU and 128G RAM. All algorithms are implemented in C++.

6. Performance evaluation

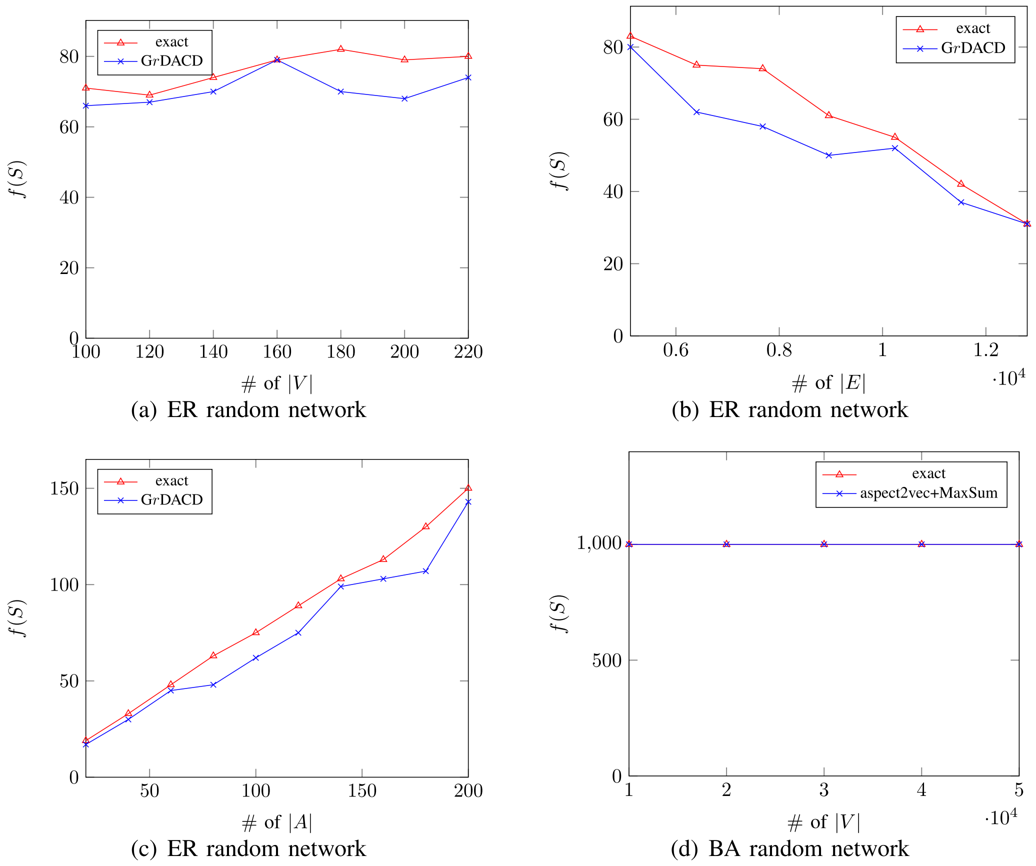
6.1. Solution quality

To study the performance of our G*r*DACD algorithm on solution quality, we develop an exact algorithm of exhaustive implementation to obtain the optimal solution in synthetic network datasets with a small number of nodes. Since the optimal solution has an exponential computational complexity, to simplify the exact algorithm design, we set $s(u) = 0$ for $\forall u \in V$, $diss(u, v) = 1$ for $\forall (u, v) \notin E$ and $r = 1$. In this setting, we only need to compare the *attribute coverage* of the result between the exact algorithm and G*r*DACD algorithm, i.e., $f(S) = |A_S|$. There are four synthetic network datasets in our experiments: (1) a set of ER random networks under different |*V*|, ranging from 100 to 220; (2) a set of ER random networks under different |*E*|, ranging from 5120 to 12800; (3) a set of ER random networks under different |*A*| ; (4) a set of BA scale free networks under different |*V*|. As shown in Fig. 3, we can clearly observe that our approximate solutions of G*r*DACD is tightly close to the exact solution in all the cases. It also can be seen in Fig. 3(d) that, in scale free networks G*r*DACD algorithm always output an optimal solution with the same *f*(*S*) as the exact solution. Based on the results on synthetic network datasets, we conclude that our G*r*DACD algorithm has an excellent performance on solution quality in the synthetic network datasets especially in the scale free networks.

6.2. Diversification performance

In this subsection, we evaluate the search result novelty of GACD and G*r*DACD under three diversity metrics defined above. We run all the algorithms on the three real-world network datasets with varying k values ($k \in [10, 100]$) to get diverse result sets. In each run, the selected algorithm obtains a result set S from a same

[2] http://snap.stanford.edu/data/.
[3] http://www.ccf.org.cn/sites/paiming/2015ccfmulu.pdf.
[4] https://aminer.org/AMinerNetwork.



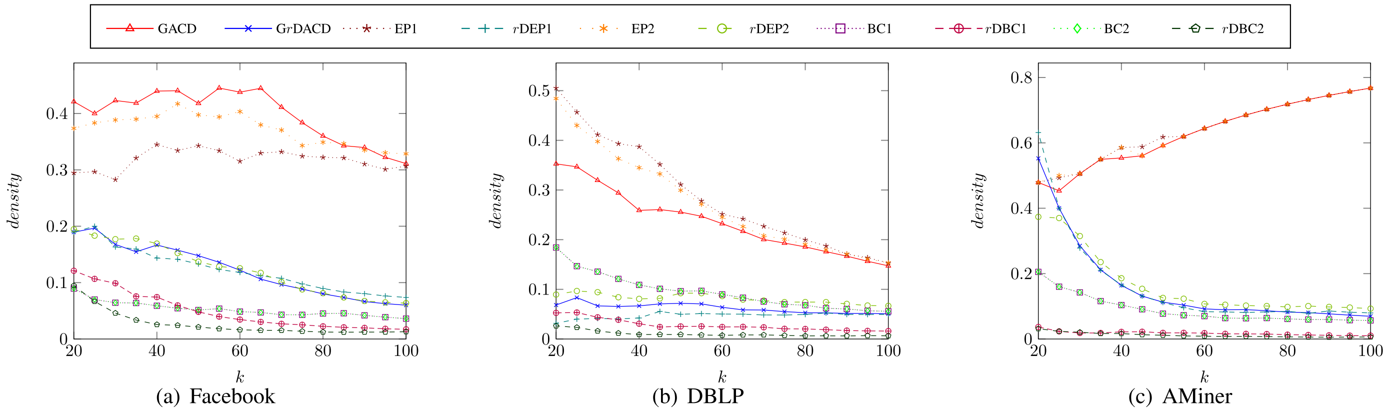
(a) ER random network

(b) ER random network

(c) ER random network

(d) BA random network

Fig. 3. Solution quality of GrDACD on synthetic networks.



(a) Facebook

(b) DBLP

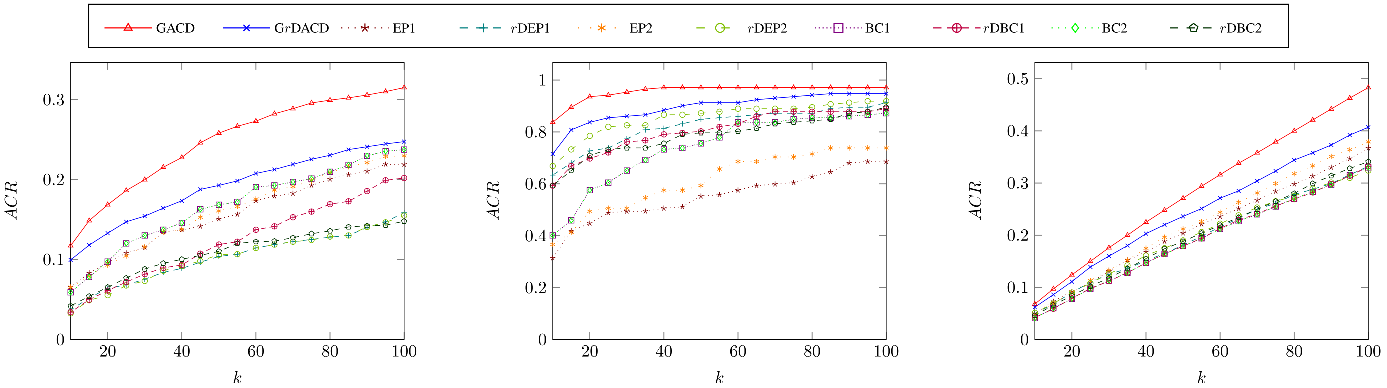
(c) AMiner

Fig. 4. Comparison on density.

query q, where $S \in V$, $|S| = k$. The *density, ARC* and *MinDiss* measures are computed on *S*, and the average of each measure is displayed for different *k* values, respectively.

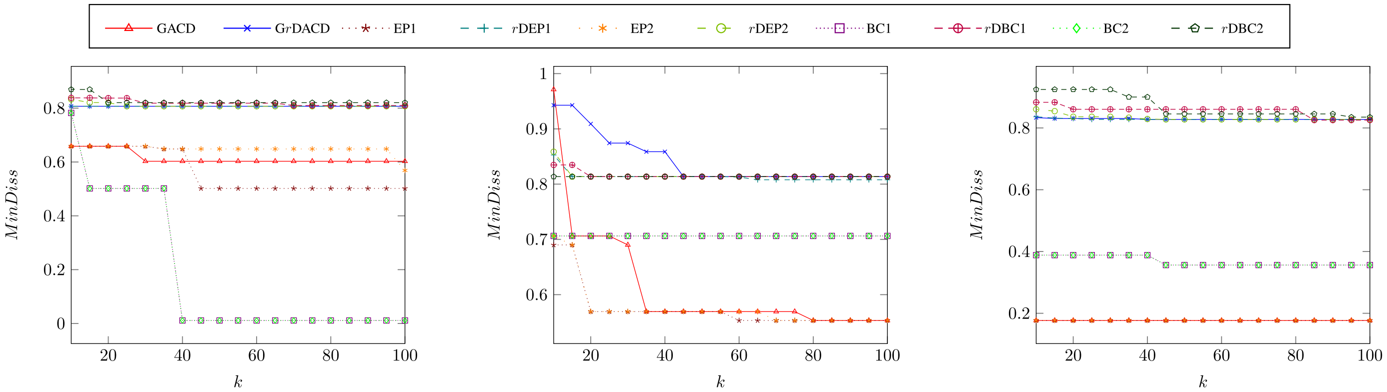
We plot the density results in Fig. 4. In general, it is clear that the three algorithms (ACD, EP1, EP2) without dissimilarity constraint score significantly higher density under most cases of *k* than other algorithms. The *expanded relevance* based algorithms always obtain a result with a lower density than other types of algorithms. Moreover, we observe that among them *r*-DBC1 and *r*-DBC2 always get a lower density than BC1 and BC2. Recall that a lower density indicates less similar to each other, which represents more diverse in structural topology to some extent. So the density results show that our GrDACD algorithm can effectively capture the structural topology diversity of the search result, and adding dissimilarity constraint can effectively eliminate redundancy of the search result.

Fig. 5 compares the performance on *attribute coverage ratio* (ACR). In general, the ACR values of all the algorithms get higher as *k* increases in all datasets. GACD shows extraordinary performance for all the cases and GrDACD achieves the second best. In DBLP social network, the search result of GACD covers all the 172 attributes of the network when *k* is greater than 35, meaning that these result authors are of strong representation, since they have published papers in all the top 172 conferences. By contrast, the ARC performance of other baselines is relatively poor. This is because there is no explicit optimization objective concerning about the attribute diversification in these models. Recall that the higher value of ACR means more likelihood that the search result can cover the latent



(a) Facebook (b) DBLP (c) AMiner

Fig. 5. Comparison on attribute coverage ratio.



(a) Facebook (b) DBLP (c) AMiner

Fig. 6. Comparison on*MinDiss*.

intent of the query. This evaluation provides a validation that our GACD and GrDACD can effectively obtain the attributed diversified search result.

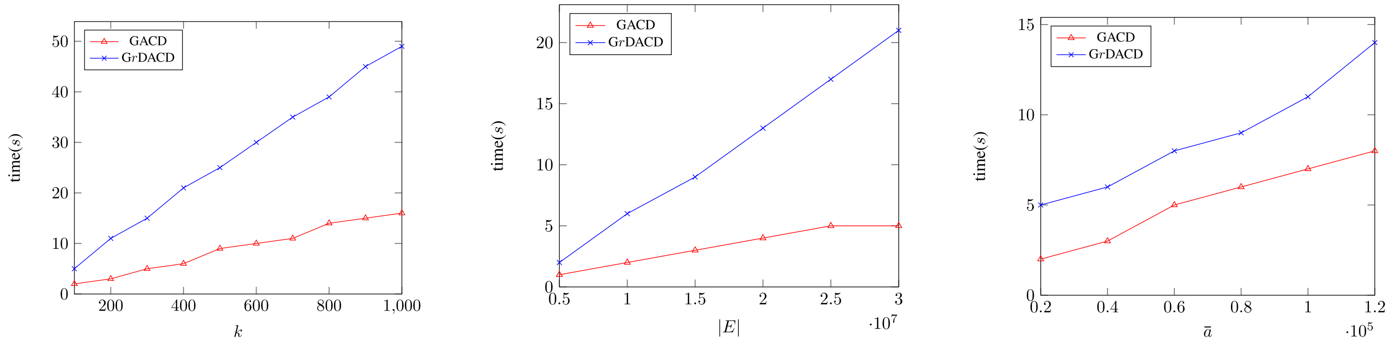
To further validate the effectiveness of the dissimilarity constraint, we report the *MinDiss* performance for these algorithms in Fig. 6. On Facebook, DBLP and AMiner datasets, we can clearly see that search results from the *dissimilarity-constrained* algorithms always outperform their *non-dissimilarity-constrained* algorithms in terms of the *MinDiss* evaluation metric. This advantages are more obvious on the AMiner network, the *dissimilarity-constrained* algorithms almost always return the results having twice higher *MinDiss* score than the *non-dissimilarity-constrained* algorithms. Our GrDACD method always perform the best or the second best in the three datasets. On the other hand, the *MinDiss* values of BC1 and BC2 are even near zero by k from 40 to 100 in Facebook network, which means that there exist at least two nodes that are extremely similar to each other in the result set. Their results are shown that they are unable to eliminate the redundancy among the search result. This is because neighbor expansion based methods tend to find the centrality or prestige nodes rather than nodes with high diversity (both structure and attribute) and novelty among them. Hence, we believe that the strong overall performance of GrDACD is due to the fact that it not only focuses on improving the diversity of node attributes but also explicitly tries to eliminate the redundancy.

6.3. Scalability

To evaluate the scalability performance of our proposed algorithms, we execute our two algorithms on a series of ER random networks over different result size k, edge size $|E|$ and average attribute size $\bar{a}$. With the runtime experiments shown in Fig. 7, we can clearly see that both GACD and GrDACD scale linearly w.r.t k, $|E|$ and $\bar{a}$. This confirms our time complexity analysis in the previous sections. In general, GACD performs better than GrACD in all the cases. GrACD only needs about 5 s to perform top 100 diversified node search in a network with 10 million edges, indicating that it is scalable to very large attributed networks.

6.4. Case study

Now we provide a case study to illustrate the effectiveness of the proposed methods. Suppose we want to search for a researcher's profile on an academic social network. The system would recommend us a limited-size list of related researchers. The query researcher might across multiple academic fields, so the recommendation list can be diverse. We simulate this recommendation based on DBLP network. Table 2 shows an example of top-5 diversified search results for Jiawei Han in DBLP network. We also count the number of conferences (attributes) that the top-5 authors covered and the number of research areas that these conferences belong to. As shown in Table 2, different search methods present very different results. The result authors found by Panther have the highest relevance to the query author, but they cover the least conferences and research areas. On contrast, our GACD and GrDACD achieve the best and second best performances in both conference and research area coverage. And both rDEP1 and rDBC1 also obtain significantly better performance than EP1 and BC1. These results confirm not only the effectiveness of our proposed GACD and GrDACD methods, but also that combining dissimilarity constraint can enhance the performance of diversification.



(a) Effect of result size k

(b) Effect of edge size $|E|$

(c) Effect of average attribute size $\bar{a}$

Fig. 7. Scalability of our algorithms.

Table 2
Case study.

Method	Panther	ACD	r -DACD	EP1	r -DEP1	BC1	r -DBC1
TOP 5	Chi Wang 0001 Yizhou Sun Tarek F. Abdelzaher Xifeng Yan Heng Ji	Li Zhang Yizhou Sun Chi Wang 0001 Xiangyu Zhang Tarek F. Abdelzaher	Tarek F. Abdelzaher Yizhou Sun Chi Wang 0001 Li Zhang Chao Zhang	Chi Wang 0001 Tarek F. Abdelzaher Xifeng Yan Heng Ji Philip S. Yu	Xin Jin Hong Cheng Dong Wang Chao Zhang Bolin Ding	Xifeng Yan Tarek F. Abdelzaher Yizhou Sun Jing Gao Jian Pei	Philip S. Yu Thomas S. Huang Xifeng Yan Wei Liu Dong Wang
conferences	40	79	73	48	65	44	68
areas	8	10	10	8	9	8	9

7. Conclusion

In this paper, we explore a practical problem of diversifying search results in attributed networks. Based on modeling the attributed diversification problem (ACD), we formulate this problem as the r-DACD problem of combining attributed diversification with dissimilarity-constrained diversification to improve novelty of search results. We show that the r-DACD problem is hard to approximate within any constant factor, and propose two algorithms with bounded approximation ratios to solve these problems. The experimental results shows that our GrDACD algorithm performs well in synthetic networks especially in the scale free networks compare with the exact algorithm. We empirically compare our algorithms with two state-of-the-art diversified search methods, as well as their improved algorithms combining a dissimilarity constraint, in terms of both the structure diversified metrics and the attributed diversified metrics in real-world attributed networks. The results validate the effectiveness of our proposed algorithms, and confirm that combining dissimilarity constraint in diversification can significantly improve search result novelty. Our future work will be focused on how to extend the diversified search method to better capture quey intent and how to deal with dynamic structure and information diffusion.

Acknowledgment

This work is supported by National Key R & D Program of China Project #2017YFB0203201 and Australian Research Council Discovery Project DP150104871.

References

[1] A.-L. Barabási, R. Albert, Emergence of scaling in random networks, Science 286 (5439) (1999) 509–512.
[2] P. Bogdanov, A. Singh, Accurate and scalable nearest neighbors in large networks based on effective importance, in: Proceedings of the 22nd ACM International Conference on Conference on Information & Knowledge Management, ACM, 2013, pp. 1009–1018.
[3] A. Borodin, H.C. Lee, Y. Ye, Max-sum diversification, monotone submodular functions and dynamic updates, in: Proceedings of the 31st Symposium on Principles of Database Systems, ACM, 2012, pp. 155–166.
[4] N. Buchbinder, M. Feldman, J.S. Naor, R. Schwartz, Submodular maximization with cardinality constraints, in: Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SIAM, 2014, pp. 1433–1452.
[5] N. Buchbinder, M. Feldman, J. Seffi, R. Schwartz, A tight linear time (1/2)-approximation for unconstrained submodular maximization, SIAM J. Comput. 44 (5) (2015) 1384–1402.
[6] J. Carbonell, J. Goldstein, The use of mmr, diversity-based reranking for reordering documents and producing summaries, in: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 1998, pp. 335–336.
[7] T. Chakraborty, N. Modani, R. Narayanam, S. Nagar, Discern: A diversified citation recommendation system for scientific queries, in: Data Engineering (ICDE), 2015 IEEE 31st International Conference on, IEEE, 2015, pp. 555–566.
[8] S. Dughmi, T. Roughgarden, M. Sundararajan, Revenue submodularity, in: Proceedings of the 10th ACM Conference on Electronic Commerce, ACM, 2009, pp. 243–252.
[9] P. ERDdS, A. R&WI, On random graphs i, Publ. Math. Debrecen 6 (1959) 290–297.
[10] U. Feige, A threshold of ln n for approximating set cover, J. ACM (JACM) 45 (4) (1998) 634–652.
[11] U. Feige, V.S. Mirrokni, J. Vondrak, Maximizing non-monotone submodular functions, SIAM J. Comput. 40 (4) (2011) 1133–1153.
[12] S. Gollapudi, A. Sharma, An axiomatic approach for result diversification, in: Proceedings of the 18th International Conference on World Wide Web, ACM, 2009, pp. 381–390.
[13] J. Håstad, Clique is hard to approximate within $n^{1-\epsilon}$, in: Foundations of Computer Science, 1996. Proceedings., 37th Annual Symposium on, IEEE, 1996, pp. 627–636.
[14] T.H. Haveliwala, Topic-sensitive pagerank, in: Proceedings of the 11th International Conference on World Wide Web, ACM, 2002, pp. 517–526.
[15] X. Huang, H. Cheng, J.X. Yu, Dense community detection in multi-valued attributed networks, Inf. Sci. 314 (2015) 77–99.
[16] R.K. Iyer, J.A. Bilmes, Submodular optimization with submodular cover and submodular knapsack constraints, in: Advances in Neural Information Processing Systems, 2013, pp. 2436–2444.
[17] S. Jegelka, J. Bilmes, Submodularity beyond submodular energies: coupling edges in graph cuts, in: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, IEEE, 2011, pp. 1897–1904.
[18] G. Jeh, J. Widom, Simrank: a measure of structural-context similarity, in: Proceedings of the Eighth ACM SIGKDD International Conference On Knowledge Discovery and Data Mining, ACM, 2002, pp. 538–543.
[19] A. Krause, A. Singh, C. Guestrin, Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies, J. Mach. Learn. Res. 9 (2008) 235–284.
[20] O. Küçüktunç, E. Saule, K. Kaya, Ü.V. Çatalyürek, Diversified recommendation on graphs: pitfalls, measures, and algorithms, in: Proceedings of the 22nd International Conference on World Wide Web, ACM, 2013, pp. 715–726.
[21] O. Küçüktunç, E. Saule, K. Kaya, Ü.V. Çatalyürek, Diversifying citation recommendations, ACM Trans. Intell. Syst. Technol. (TIST) 5 (4) (2015) 55.

[22] A. Kulik, H. Shachnai, T. Tamir, Maximizing submodular set functions subject to multiple linear constraints, in: Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms, Society for Industrial and Applied Mathematics, 2009, pp. 545–554.

[23] P. Lee, L.V. Lakshmanan, J.X. Yu, On top-k structural similarity search, in: Data Engineering (ICDE), 2012 IEEE 28th International Conference on, IEEE, 2012, pp. 774–785.

[24] R.-H. Li, J.X. Yu, Scalable diversified ranking on large graphs, Knowl. Data Eng. IEEE Trans. 25 (9) (2013) 2133–2146.

[25] H. Lin, J. Bilmes, Multi-document summarization via budgeted maximization of submodular functions, in: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, 2010, pp. 912–920.

[26] Q. Mei, J. Guo, D. Radev, Divrank: the interplay of prestige and diversity in information networks, in: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Acm, 2010, pp. 1009–1018.

[27] D. Mottin, F. Bonchi, F. Gullo, Graph query reformulation with diversity, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2015, pp. 825–834.

[28] G.L. Nemhauser, L.A. Wolsey, M.L. Fisher, An analysis of approximations for maximizing submodular set functions, Math. Program. 14 (1) (1978) 265–294.

[29] L. Page, S. Brin, R. Motwani, T. Winograd, The pagerank citation ranking: bringing order to the web. (1999).

[30] S. Sakai, M. Togasaki, K. Yamazaki, A note on greedy algorithms for the maximum weighted independent set problem, Discrete Appl. Math. 126 (2) (2003) 313–322.

[31] P. Sarkar, A.W. Moore, Fast nearest-neighbor search in disk-resident graphs, in: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2010, pp. 513–522.

[32] H. Tong, J. He, Z. Wen, R. Konuru, C.-Y. Lin, Diversified ranking on large graphs: an optimization viewpoint, in: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2011, pp. 1028–1036.

[33] J. Vondrák, Symmetry and approximability of submodular maximization problems, SIAM J. Comput. 42 (1) (2013) 265–304.

[34] Z. Wang, J. Liao, Q. Cao, H. Qi, Z. Wang, Friendbook: a semantic-based friend recommendation system for social networks, Mobile Comput. IEEE Trans. 14 (3) (2015) 538–551.

[35] Y. Wu, R. Jin, X. Zhang, Fast and unified local search for random walk based k-nearest-neighbor query in large graphs, in: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, ACM, 2014, pp. 1139–1150.

[36] C. Zhang, L. Shou, K. Chen, G. Chen, Y. Bei, Evaluating geo-social influence in location-based social networks, in: Proceedings of the 21st ACM International Conference on Information and Knowledge Management, ACM, 2012, pp. 1442–1451.

[37] J. Zhang, J. Tang, C. Ma, H. Tong, Y. Jing, J. Li, Panther: Fast top-k similarity search on large networks, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2015, pp. 1445–1454.

[38] P. Zhao, J. Han, Y. Sun, P-rank: a comprehensive structural similarity measure over information networks, in: Proceedings of the 18th ACM Conference on Information and Knowledge Management, ACM, 2009, pp. 553–562.