Scalable Diversified Ranking on Large Graphs

Rong-Hua Li and Jeffery Xu Yu

Abstract—Enhancing diversity in ranking on graphs has been identified as an important retrieval and mining task. Nevertheless, many existing diversified ranking algorithms either cannot be scalable to large graphs due to the time or memory requirements, or lack an intuitive and reasonable diversified ranking measure. In this paper, we propose a new diversified ranking measure on large graphs, which captures both *relevance* and *diversity*, and formulate the diversified ranking problem as a submodular set function maximization problem. Based on the submodularity of the proposed measure, we develop an efficient greedy algorithm with linear time and space complexity w.r.t. the size of the graph to achieve near-optimal diversified ranking. In addition, we present a generalized diversified ranking measure and give a near-optimal randomized greedy algorithm with linear time and space complexity for optimizing it. We evaluate the proposed methods through extensive experiments on five real datasets. The experimental results demonstrate the effectiveness and efficiency of the proposed algorithms.

Index Terms—Diversified Ranking, Graph Algorithms, Scalability, Flajolet-Martin sketch, Submodular Function.

1 Introduction

Ranking nodes on graphs is a fundamental task in information retrieval, data mining, and social network analysis. It has a large number of applications such as ranking web-pages [1], measuring centrality in social networks [2], as well as enhancing personalized services for web search [3]. Most of existing graph-based ranking algorithms are based on the stationary distribution of the random walk on graphs, such as the PageRank algorithm [1] and its variants [3][4]. The idea of this random walk based ranking algorithms is that the node of a graph should be ranked higher if there are more high-ranking nodes link to it. This basic idea has become a crucial criteria for designing ranking algorithms on graphs and also has been successfully applied in many applications.

However, as discussed in [5][6], the design criteria lead to many nodes found in the top-K ranking list are similar because it only considers the relevance of the nodes. It reduces the ranking effectiveness when the applications need to incorporate diversity into the top-K ranking results. Take Flickr (http://www.flickr.com), which is a well known photo shared website, as an example. Users in Flickr can make friends and join in many interest groups. Consider a retrieval task of finding the top-K relevant users who are similar to a given user but are from as many interest groups as possible in the Flickr social network. In general, we can use personalized PageRank algorithms [1][3][4] to rank the users, and then find the top-K users based on their personalized PageRank scores. However, the top-K users found by the personalized PageRank typically include many users who are in the same interests group, thereby they cannot meet our objective of diversity. To this end, we need to take the diversity of the top-K ranking list into account for designing ranking algorithms. In other words, the ranking algorithms in this case should produce diversified ranking results so as to cover as many groups as possible.

Recently, improving diversity in top-K ranking results has attracted much attention as it has a variety of applications in information retrieval and data mining areas. There exists a large body of work on search results diversification both in text and graph datasets respectively. In this paper, we focus on enhancing diversity in ranking on graph datasets. We are interested in finding the top-K nodes that are not only relevant to the query but also dissimilar to one another. Here the relevance of the nodes is measured by their personalized PageRank scores.

In the literature, there are four frameworks for diversified ranking on graphs. The first one is based on a greedy vertex selection procedure [5][7], the second one is based on a so-called vertex reinforced random walk [6], the third framework is based on optimizing the predefined diversified measures [8][9], and the last one is based on the resistive graph centers [10]. In particular, the greedy vertex selection procedure chooses a vertex with a maximum random walk based ranking score at a time, and then removes the selected vertex from the graph. To get the top-K ranking list, this process repeats K times. To the best of our knowledge, there are two algorithms based on this framework: the Grasshopper algorithm [5] and the manifold rank with stop points algorithm [7]. Both algorithms have empirically shown that they can improve diversity in ranking on graph data. However, the major drawback of this type of algorithms is that they have cubic time complexity, thus they cannot be scalable to large graphs. Another drawback of this type of algorithms is that they lack a theoretical explanation

- *The Chinese University of Hong Kong,*
  E-mail: {rhli,yu}@se.cuhk.edu.hk

for the algorithms why they can improve diversity in ranking results. Some improvements of this point have been achieved in the second framework [6]. In [6], Mei, et al. propose a diversified ranking algorithm, called DivRank, based on a vertex reinforced random walk, and present an optimization explanation for DivRank to improve diversity in ranking. However, the explanation is only suitable for undirected graphs. In addition, the convergence property of DivRank is not clear, because it resorts to some approximation strategies to the original vertex reinforced random walks. Another drawback of DivRank is that it cannot be scalable to large graphs for two reasons. On one hand, DivRank dynamically updates the transition matrix at each iteration. This procedure may result in a full transition matrix, thus it cannot be stored in main memory if the graph is very large. On the other hand, the full transition matrix increases the computational cost for the matrix-vector multiplication. Tong, et al. in [8] propose a scalable diversified ranking algorithm by optimizing a predefined diversified ranking measure. However, the motivation of their diversified ranking measure is not explicitly clarified. Specifically, for measuring diversity, their measure is based on a multiplication of the so-called 'Google matrix' and the personalized PageRank vector, which lacks a clear topological explanation. Hence, it does not directly reflect diversity of a set of nodes from graph structural perspective. The last notable diversified ranking algorithm is based on resistive graph centers [10]. Similar to the greedy vertex selection algorithms, the time complexity of this algorithm is cubic, thus it cannot scale to large graphs.

To overcome the problems in the existing algorithms, in this paper, we present a novel diversified ranking method on graphs. The basic idea of our approach is that we first calculate the personalized PageRank vector on the basis of the query node, and then perform a carefully designed vertex selection algorithm to find the top-K diversified ranking list according to a predefined diversified ranking measure. The key challenges in our method are (1) how to define an intuitive and reasonable diversified ranking measure that captures both relevance and diversity, and (2) how to develop an efficient vertex selection algorithm to optimize the diversified ranking measure. To this end, firstly, we propose a modified definition of *expansion* on graph to capture the diversity of the nodes. The key intuition is that if the nodes have large *expansion*, then the nodes will be dissimilar to each other, thus leading to diversity. Secondly, based on this definition, we propose a novel diversified ranking measure by combining relevance and diversity. We show that the proposed measure is a nondecreasing submodular set function. Based on the submodularity of the proposed measure, we design an efficient greedy algorithm with linear time and space complexity w.r.t. the size of the graph to find the top-K diversified ranking list. Thirdly, we further present a generalized diversified ranking measure based on the definition of *k-step expansion*, and propose a randomized greedy algorithm with linear time and space complexity to optimize it accurately. Finally, we compare our proposed methods with six existing algorithms on five real networks. The experimental results demonstrate the effectiveness, efficiency and scalability of the proposed algorithms. The preliminary study of this work is reported in [9].

The rest of this paper is organized as follows. We give a briefly review of personalized PageRank algorithm and present our new diversified ranking measure as well as our problem formulation in Section 2. We show the submodularity of the proposed measure and give a near-optimal greedy algorithm for finding top-K diversified ranking in Section 3. We present a generalized diversified ranking measure and a randomized greedy algorithm in Section 4. Extensive experiments are reported in 5, and related work is discussed in Section 6. We conclude this work in Section 7.

2 PRELIMINARIES

In this section, we first briefly review the personalized PageRank algorithm that is used as a basic measure of relevance in diversified ranking on graphs. Then, we propose a new diversified ranking measure and formulate our diversified ranking problem as a discrete optimization problem.

2.1 Personalized PageRank algorithm

Personalized PageRank [1][3][4] is a well known approach for query-dependent ranking on graphs, and it has been successfully used in various applications in the past decades. We briefly describe the personalized PageRank algorithm below.

Given a query vector r (also call teleport vector in many literature [10]), and a graph G. Then, the personalized PageRank vector w can be calculated by the following iterative equation:

$$w = (1-\alpha)r + \alpha A^T w, \tag{1}$$

where α is a damping factor, and A is the adjacency matrix of graph G. The iterative equation in Eq. (1) can converge to a fixed point, which corresponds to the stationary distribution of the Markov chain. The resulting vector w will be utilized to rank the nodes of the graph.

However, the personalized PageRank does not consider diversity of the ranking results. This is because the personalized PageRank makes use of the stationary distribution of the random walks for ranking nodes in graph. The random walk on graph can form a Markov chain. By the fundamental theorem of Markov chain [11], the stationary distribution of the walks is inversely proportional to the hitting time. If a node is hit very frequently by random walks, then the node will have a high personalized PageRank score. Also, if a node is hit frequently, all its neighbors are most likely to be hit frequently, thus its neighbors also get high personalized PageRank scores. Obviously, this process spreads to many adjacent nodes in the top-K ranking results. In other words, the top-K ranking list found by the personalized PageRank may contain many similar nodes, which reduces the ranking effectiveness in the applications that need to incorporate diversity.

2.2 Problem formulation

In the literature, there are many ranking algorithms on graphs [5][6][7] that aim at improving diversity. However, as our analysis given in the introduction, the existing diversified ranking algorithms either cannot scale to large-scale graphs or lack an intuitive and reasonable diversified ranking measure. To this end, in this paper, we propose a new diversified ranking measure on graphs and design a scalable algorithm for optimizing it accurately. Below, we first give some important notations and definitions, and then formulate our diversified ranking problem.

Notations and definitions: Consider a graph $G = (V, E)$, with a set of nodes V and a set of edges E, where the size of nodes is $n = |V|$.

Definition 2.1: Let S be a set of nodes. The *expanded set* of S is denoted by $N(S)$ such that $N(S) = S \cup \{v \in (V - S)|\ \exists u \in S, (u, v) \in E\}$. The *expansion* of a set of nodes, S, is the size of the expanded set, $N(S)$, denoted as $|N(S)|$. And the *expansion ratio* is defined as $\sigma = |N(S)|/n$.

It is worth mentioning that our definition of expansion is based on the topological structure of the graph. which can be either undirected or directed. In addition, it is important to note that our definition of expansion is different from the definition of expansion given in the expander graph [12] where the expansion of a graph equals to the minimum expansion ratio among all the expanded sets.

With Def. 2.1, a set of nodes with a large expansion ratio implies that the nodes are dissimilar to one another. Here, the intuition behind is that two nodes are dissimilar if they do not share the common neighbors in a graph. The larger expansion ratio the set of nodes has, the better diversity among the set of nodes they can achieve. Consider a graph in Fig. 1(a). Assume we select three nodes (red nodes) in Fig. 1(b) and Fig. 1(c), respectively. Then, the expansion ratio of the selected nodes in Fig. 1(b) and Fig. 1(c) are 0.6 and 0.9 respectively. The selected nodes in Fig. 1(b) are well connected, thus they can be similar to one another. On the other hand, there is no edge between any two selected nodes in Fig. 1(c), thus they can be dissimilar to each other. As a result, the selected nodes in Fig. 1(c) are more diverse than the selected nodes in Fig. 1(b). This example indicates that nodes with a larger expansion ratio result in better diversity. Our diversified ranking measure is based on this key intuition.

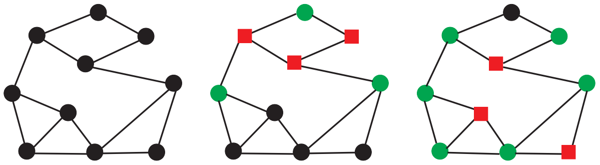

(a) A graph G (b) $\sigma = 0.6$ (c) $\sigma = 0.9$

Fig. 1. Illustration of our idea: expansion ratio vs diversity. Red square nodes denote the selected nodes and green nodes are the expanded nodes (color online).

Diversified ranking measure: The most commonly used criteria for combining relevance and diversity are the so-called maximum marginal relevance (MMR) [13], which is a linear combination of relevance and diversity and is widely used in many document retrieval systems. With MMR, a document that has a high marginal relevance means that it is relevant to the query and is dissimilarity to the previously selected documents. Similarly, in a graph, a node with a high diversified rank should (1) have a high personalized PageRank score, and (2) be dissimilar to the other selected nodes. Our definition of expansion ratio can be deemed as a diversity measure. And we aim at finding a subset S of K nodes such that (1) the nodes in S have high personalized PageRank scores and (2) the expansion ratio of $|N(S)|/n$ is maximum. Formally, our goal is to maximize the following diversified ranking measure:

$$F(S) = (1 - \lambda) \sum_{u \in S} w_u + \lambda \frac{|N(S)|}{n}, \tag{2}$$

where w_u denotes the personalized PageRank score of node u, and $\lambda \in [0, 1]$ is a parameter that is used to tradeoff relevance and diversity. The first term in Eq. (2) is the sum of the personalized PageRank scores over the ranking results, which reflects the relevance of the ranking results. The second term is the expansion ratio of the ranking results. As discussed, a better expansion ratio implies better diversity. Hence, Eq. (2) captures both relevance and diversity.

Note that $F(S)$ does not consider the ordering of the top-K ranking list. This is because our definition is based on a mild assumption that the users in a real retrieval system generally focus on all the top-K results. This assumption is typically reasonable in many practical applications [5][6][7]. However, in Section 3.3, we will show that our proposed algorithm still yields an ordering results based on both relevance and diversity score of the node.

To summarize, our problem of finding top-K diversified ranking on graph is formalized as follows:

$$\begin{aligned} \arg\max_{S \subseteq V} \quad & F(S) \\ s.t. \quad & |S| = K. \end{aligned} \tag{3}$$

3 Diversified ranking algorithm

As discussed, our diversified ranking problem is to maximize the proposed diversified ranking measure subject to a cardinality constraint (Eq. (3)). The following theorem shows that the problem formulated in Eq. (3) is NP-hard in general graphs.

Theorem 3.1: *For a general graph $G = (V, E)$, the optimization problem in Eq. (3) is NP-hard.*

Proof Sketch: We consider a special case of our problem defined in Eq. (3) and show it is NP-hard. Let $\lambda = 1$, then the problem is equal to maximize $|N(S)|$ subject to $|S| = K$. This special problem is equivalent to the maximal expansion problem defined in [14] which is known to be NP-hard. As a consequence, our problem defined in Eq. (3) is also NP-hard. □

Given the hardness of our problem, there is no hope to optimally solve the top-K diversified ranking problem on general graphs in polynomial time unless P=NP. Only on trees, the diversified ranking problem (Eq. (3)) can be solved optimally in polynomial time by a dynamic programming algorithm, which we describe in the following subsection.

3.1 Diversified ranking on trees

Although the diversified ranking problem on general graphs is NP-hard, we show that it can be solved optimally in polynomial time when the graph is a tree. Our polynomial-time algorithm is based on dynamic programming. The basic idea is described as follows.

Consider a subtree whose root has x children, the optimal way of finding K nodes from the subtree for the diversified ranking list must follow one of two cases. In the first case, we include the root of the subtree to the ranking list and then recurse on the children with a budget of K-1. In the second case, we do not add the root of the subtree, and instead recurse on the children with a budget of K. A naive implementation of this recursion needs to partition x children into K (or $K - 1$) parts in all possible ways. Obviously, this is extremely expensive if $x \gg 2$. To overcome this, we construct a transformation that converts the general tree to a binary tree without altering optimum. The transformation is described as follows.

We start from the root of tree T, denoted by $root(T)$. Assume u is an internal node of T with children $u_1, u_2, \cdots, u_x$ and $x > 2$. Then, we replace u by a binary tree with depth at most $\log_2 x$ and leaves $u_1, u_2, \cdots, u_x$. In particular, let u_1 be left child of u. Add a new node z_1 and let it be the right child of u. Then, let the remainder children of u be the children of z_1. Repeat these steps until every nodes have at most two children. We set the personalized PageRank score of the newly added nodes to $-\infty$ and the personalized PageRank score of $u, u_1, u_2, \cdots, u_x$ are the same as before. This can ensure that the newly added nodes will never be added into the top-K ranking list. Obviously, the depth of the new tree (a binary tree) is at most a factor of $\log_2 d_{\max}$ larger than the depth of the original tree. Here $d_{\max}$ denotes the maximum out-degree of a node in the original tree. Further, the size of the binary tree is at most twice the size of the original tree. More importantly, it is not very hard to verify that the optimal solution of Eq. (3) on the binary tree is the same as the optimal solution on the original tree. Similar constructions have been used for various applications [15][16].

Based on this construction, we can assume the tree is binary, and is denoted by $\mathcal{T}$. For each node u in $\mathcal{T}$, we define a cost function w.r.t. the current solution S as $C(u, S) = (1 - \lambda)w_u + \lambda|N(\{u\}) - N(S)|/n$. Let $F(u, S, k)$ be the optimal solution in the subtree rooted by u with budget k, where the set S maintains the current solution. And let $l(u)$ $(r(u))$ denotes the left (right) child of node u. Then, the recursive equation of the dynamic programming (DP) is given by

$$\begin{aligned}&F(u,S,k) = \max\{\\&\max_{i=0}^{k}\{F(l(u),S,i) + F(r(u),S,k-i)\},\\&C(u,S) + \max_{i=0}^{k-1}\{F(l(u),S\cup\{u\},i)\\&+F(r(u),S\cup\{u\},k-1-i)\}\}.\end{aligned}$$

The first term of the recursive equation corresponds to do not select u to be in S and the second term corresponds to add u into S. We analyze the time complexity of the DP algorithm as follows (here we use a budget of K). First, building the binary tree takes $O(n \log_2 d_{\max})$ time. Second, we need to evaluate the recursion $O(K)$ times for each node in the binary tree. For each such evaluation, it takes $O(K)$ time. Notice that computing $C(u, S)$ can be done in constant time in a binary tree. There are $O(n \log_2 d_{\max}))$ nodes in the binary tree. Putting all it together, the time complexity of the DP algorithm is $O(K^2 n \log_2 d_{\max}))$.

3.2 Submodularity

Since the diversified ranking problem on general graphs is NP-hard, we resort to develop approximate algorithms for solving it efficiently. Below, we prove that our proposed diversified ranking measure ($F(S)$) is a nondecreasing submodular set function, which allows us to develop a near-optimal greedy algorithm for maximizing it efficiently. We give the definition of the nondecreasing submodular set function [17] as follows.

Definition 3.1: Let V be a finite set, a real valued function $f(S)$ on the set of subsets of V, S, is called a nondecreasing submodular set function, if the following conditions hold.

- **Nondecreasing**: For any subsets S and T of V such that $S \subseteq T \subseteq V$, we have $f(S) \leq f(T)$.
- **Submodularity**: Let $\rho_j(S) = f(S \cup \{j\}) - f(S)$ be the marginal gain. Then, for any subsets S and T of V such that $S \subseteq T \subseteq V$ and $j \in V \backslash T$, we have $\rho_j(S) \geq \rho_j(T)$.

We prove that Eq. (2) is a nondecreasing submodular function with $F(\emptyset) = 0$, where $\emptyset$ is an empty set. We state the theorem as follows.

Theorem 3.2: *The set function $F(S)$ defined in Eq. (2) is a nondecreasing submodular function with $F(\emptyset) = 0$.*

Proof: For $\forall S \subseteq T \subseteq V$ and $j \in V \backslash T$, let $\rho_j(S) = F(S \cup \{j\}) - F(S)$, and $\rho_j(T) = F(T \cup \{j\}) - F(T)$. Then, we have

$$\begin{aligned} \rho_j(T) &= (1-\lambda)w_j + \lambda \tfrac{|N(T \cup \{j\})| - |N(T)|}{n} \\ &= (1-\lambda)w_j + \lambda \tfrac{|N(\{j\}) - N(T)|}{n} \\ &\geq 0. \end{aligned} \tag{4}$$

Note that the nondecreasing property of $F(S)$ can be guaranteed by $\rho_j(T) \geq 0$.

Similarly, we have $\rho_j(S) = (1-\lambda)w_j + \lambda \frac{|N(\{j\}) - N(S)|}{n} \geq 0$. By definition, we have $F(\emptyset) = 0$ and $|N(\{j\}) - N(S)| \geq |N(\{j\}) - N(T)|$. Hence, we conclude $\rho_j(S) \geq \rho_j(T) \geq 0$. This completes the proof. □

3.3 The greedy algorithm

Because our diversified ranking measure exhibits submodularity property, with the founding in [17], we develop an efficient greedy algorithm with a $1 - 1/e$ approximation guarantee for our top-K diversified ranking problem. Alg. 1 outline our greedy algorithm.

Algorithm 1 The Greedy Algorithm

```
Input:  Graph $G = (V, E)$, K, damping factor $\alpha$,
        adjacency matrix $A$, teleport vector $r$,
        and parameter $\lambda$
Output: A set $S$ with K nodes

 1: Compute the personalized PageRank vector $w$;
 2: Initialize the answer set $S \leftarrow \emptyset$;
 3: For each node $v_i$, initialize an indicator array Expan[i]
    $\leftarrow 0$;
 4: for iter = 1 to K do
 5:   $max \leftarrow -1$;
 6:   $maxIdx \leftarrow 0$;
 7:   for each node $v_i \in (V - S)$ do
 8:     $counter \leftarrow 0$;
 9:     for each neighbor node $(v_j)$ of $v_i$ do
10:       if Expan[j] = 0 then
11:         $counter \leftarrow counter + 1$;
12:     if $((1-\lambda)w_i + \lambda \times counter/|V|) > max$ then
13:       $max \leftarrow (1-\lambda)w_i + \lambda \times counter/|V|$;
14:       $maxIdx \leftarrow i$;
15:   $S \leftarrow S \cup \{v_{maxIdx}\}$;
16:   for each neighbor node $(v_j)$ of $v_{maxIdx}$ do
17:     Expan[j] $\leftarrow 1$;
18: return $S$;
```

In Alg. 1, the algorithm first computes the personalized PageRank vector as the initial ranking (line 1), which measures the relevance of the nodes. Then, in each iteration, the algorithm chooses a node u with the maximum marginal gain $\rho_u(S) = (1-\lambda)w_u + \frac{\lambda}{n}|N(\{u\}) - N(S)|$ (line 7-15), and adds it into the answer set S. To get the top-K ranking list, this procedure will repeat K times (line 4-17). The algorithm will produce an ordering ranking list according to $\rho_u(S)$. Since $\rho_u(S)$ satisfies the nondecreasing properties, Alg. 1 will output a reasonable ranking such that the node with a high ranking score will appear in the top ranking list.

Theoretically, the following theorem shows that Alg. 1 obtains a near-optimal solution.

Theorem 3.3: *Alg. 1 is a $1-1/e$ approximation algorithm for the top-K diversified ranking problem (Eq. (3)).*

Proof Sketch: This can be proved by a similar argument that has been used to prove the approximation factor of the greedy algorithm for submodular set function maximization problem [17]. □

It is worth mentioning that the $1 - 1/e$ approximation factor is tight [18]. In other words, there are no other polynomial-time algorithms that can achieve a more tight approximation factor unless P=NP. Below, we analyze the time and space complexity of Alg. 1.

Complexity analysis of the greedy algorithm: The time complexity of Alg. 1 is $O(K|E|)$. Specifically, in line 1, Alg. 1 takes $O(|E|)$ time to compute the personalized PageRank vector. The time complexity from line 4 to line 17 is $O(K|E|)$. This is because the algorithm needs to visit all the nodes and their corresponding neighbors, and the total number of nodes visiting by the algorithm equals to $2|E|$ in the worse-case. Moreover, we can use the so-called CELF framework to accelerate Alg. 1, which will result in several times speedup [19]. For the space complexity, Alg. 1 needs to store the input graph G, the personalized PageRank vector w, the answer set S, and an indicator array, which lead to $O(|V| + |E|)$ in total. Put it all together, the algorithm has linear time and space complexity w.r.t. the graph size, and thus it can be scalable to large-scale graphs.

3.4 Connection to dominating set problem

The proposed top-K diversified ranking problem (Eq. (3)) is well connected to the dominating set problem in graph theory [20]. The minimum dominating set problem in graph theory aims to find the minimum number of nodes whose expanded set can cover the whole graph. In other words, the nodes in the minimum dominating set can dominate the other nodes of the graph. The domination number (DN) of a graph is the cardinality of the minimum dominating set. It is well known that the minimum dominating set problem is NP-hard. There is an efficient greedy algorithm with $1 + ln(|V|)$ approximation factor to compute the DN and the dominating set of a graph [20]. Specifically, the greedy algorithm chooses a node with the maximal marginal gain ($\rho_u(S) = |N_u(S \cup \{u\})| - |N_u(S)|$) at a time, and it terminates when the expanded set of the selected nodes cover the whole graph. Note that the minimum dominating set problem only considers the expansion of the nodes and ignore the relevance of the nodes, thus cannot be directly applied to our problem. Moreover, our top-K diversified ranking problem aims to find the K nodes such that they are relevant to the query and simultaneously dissimilar to one another, and it is not to find the minimum number of nodes such that their expanded set can cover the whole graph.

In the case that K exceeds the dominance number (DN) of the graph, Alg. 1 will choose nodes in terms of their personalized PageRank scores. However, in many real graphs, K is significantly smaller than the DN of the graph. We will address this point in our experimental studies in Section 5.

4 Generalized Diversified Ranking

In this section, we first propose a generalized diversified ranking measure, and design an efficient greedy algorithm for optimize it accurately. Then, we discuss other potential variants of our diversified ranking measures.

4.1 Generalized diversified ranking measure

The proposed diversified ranking measure ($F(S)$) in Def. 2.1, only considers the immediate neighborhood information of S. Naturally, we can generalize the diversified ranking measure $F(S)$ by taking the k-step nearest neighbors into account.[1] We call such a measure a generalized diversified ranking measure and denote it by $F_k(S)$. In the following, we first give the definitions of k-step expanded set and k-step expansion.

Definition 4.1: Let S be a set of nodes. The *k-step expanded set* of S is denoted by $N_k(S)$ such that $N_k(S) = S \cup \{v \in (V - S) | \exists u \in S, d(u,v) \leq k\}$, where $d(u,v)$ denotes the length of the shortest path from u to v. The *k-step expansion* of S is the cardinality of the k-step expanded set denoted as $|N_k(S)|$. And the *k-step expansion ratio* is defined as $\sigma_k = |N_k(S)|/n$.

Based on the k-step expansion, we define the generalized diversified ranking measure $F_k(S)$ as follows.

$$F_k(S) = (1-\lambda)\sum_{u \in S} w_u + \lambda \frac{|N_k(S)|}{n} \tag{5}$$

Obviously, $F(S)$ is a special case of $F_k(S)$ when $k = 1$. Like $F(S)$, $F_k(S)$ is also a nondecreasing submodular function. We give a theorem as follows. The proof is similar to the proof of Theorem 3.2, thus we omit it for brevity.

Theorem 4.1: *The set function $F_k(S)$ defined in Eq. (5) is a nondecreasing submodular function with $F_k(\emptyset) = 0$, where $\emptyset$ denotes an empty set.*

Likewise, the problem of maximizing the set function $F_k(S)$ subject to a cardinality constraint is NP-hard. However, based on the submodularity property of $F_k(S)$, we can develop a greedy algorithm to optimize it accurately. Now, the problem is that the greedy algorithm needs to find a node with the maximum marginal gain $\rho_u(S) = (1-\lambda)w_u + \frac{\lambda}{n}|N_k(\{u\}) - N_k(S)|$ in each iteration. Unlike Alg. 1, the marginal gain $\rho_u(S)$ cannot be calculated in linear time complexity when $k > 1$. A naive implementation of maximizing $F_k(S)$ is described as follows. First, we construct a new graph such that any two nodes u and v of the new graph have an edge (u,v) if u can reach v in k ($k > 1$) hops in the original graph. Then, we perform Alg. 1 on the new graph. The construction of the new graph can be implemented by Floyd algorithm [21], resulting in $O(|V|^3)$ time complexity. And performing Alg. 1 on the new graph will take $O(K|E|')$ time complexity, here $|E|'$ denotes the number of edges in the new graph. Hence, the time complexity of this naive algorithm is $O(|V|^3)$, which is clearly not scalable. In the following, we develop a randomized greedy algorithm with linear time complexity using the Flajolet-Martin (FM) sketch [22].

4.2 The randomized greedy algorithm

Recall that the major time-consuming step for optimizing the generalized diversified ranking measure (Eq. (5)) is to evaluate the marginal gain ($\rho_u(S) = (1-\lambda)w_u + \frac{\lambda}{n}|N_k(S \cup \{u\}) - N_k(S)|$). Inspired by the idea of approximate neighbor function [23], we propose a randomized greedy algorithm for the generalized diversified ranking problem using the FM sketch. The FM sketch is a probabilistic counting structure, which can be used to estimate the number of distinct elements (cardinality) in a multi-set [22]. Assume the cardinality of a multi-set A is C, then the FM sketch only uses $\log C + t$ bits for estimating C in high accuracy, where t is a small constant. More specifically, the FM sketch is a bitmap with size $s = \log C + t$. There is a hash function $h : A \rightarrow \{1, \cdots, s\}$, which maps an element a in A to a bit $i = \{1, \cdots, s\}$ in the bitmap with probability $\Pr(h(a) = i) = 1/(2^{i+1})$. Initially, all bits in the bitmap is set to 0. Then, each element $a \in A$ is inserted into the bitmap by setting the corresponding $h(a)$-th bit to 1. Finally, an asymptotically unbiased estimation of the cardinality C can be obtained by $2^c/0.77351$, where c denotes the position of the least-significant zero bit in the bitmap. We can use multiple hash functions to boost the estimating accuracy. For the sake of brevity, we only consider one hash function to illustrate the algorithm. In addition, an important property of the FM sketch is that it can be easily applied to estimate the cardinality of the union of two multi-sets if these two multi-sets come from the same domain. In particular, we can construct a FM sketch with the same size for each multi-set. To estimate the cardinality of the union of two multi-sets, we only need to do a bitwise-OR between the two FM sketches, and then estimate the cardinality based on the resulting FM sketch.

[1] Here, we use small letter k to distinguish K which is used to denote the cardinality of our top-K ranking results.

Algorithm 2 The Randomized Greedy Algorithm

```
Input:  Graph $G = (V, E)$, K, damping factor $\alpha$,
        adjacency matrix $A$, teleport vector $r$,
        parameter $k$ of the k-step expansion,
        parameter $\lambda$
Output: A set $S$ with K nodes
 1: Compute the personalized PageRank vector $w$;
 2: Let $h : \{v_1, \cdots, v_n\} \rightarrow \{1, \cdots, s\}$ be the hash function
    that maps the nodes to a position of the BITMAP, here
    $s$ is the size of the BITMAP ;
 3: for each node $v_i \in V$ do
 4:     Initialize a BITMAP FM[i] $\leftarrow 0$;
 5:     Set the $h(v_i)$-bit of FM[i] to 1;
 6:     Initialize a temporary BITMAP TFM[i] $\leftarrow 0$;
 7: for iter = 1 : k do
 8:     for each node $v_i \in V$ do
 9:         TFM[i] $\leftarrow$ FM[i];
10:     for each edge $(v_i, v_j) \in E$ do
11:         FM[i] = (FM[i]) BITWISE-OR (TFM[j]);
12: Initialize the answer set $S \leftarrow \emptyset$;
13: Initialize two BITMAPs NBP $\leftarrow 0$, OBP $\leftarrow 0$;
14: $c \leftarrow 0$;
15: for iter = 1 to K do
16:     $max \leftarrow -1$;
17:     $maxIdx \leftarrow 0$;
18:     for each node $v_i \in (V - S)$ do
19:         OBP $\leftarrow$ (NBP) BITWISE-OR (FM[i]);
20:         Let $t$ be the position of the right most 0 bit in the
            BITMAP OBP;
21:         $counter \leftarrow 2^t/0.77351$;
22:         $counter \leftarrow counter - c$;
23:         if $(1 - \lambda)w_i + \lambda \times counter/|V| > max$ then
24:             $max \leftarrow (1 - \lambda)w_i + \lambda \times counter/|V|$;
25:             $maxIdx \leftarrow i$;
26:     $S \leftarrow S \cup \{v_{maxIdx}\}$;
27:     NBP $\leftarrow$ (NBP) BITWISE-OR (FM[maxIdx]);
28:     Let $t$ be the position of the right most 0 bit in the
        BITMAP NBP;
29:     $c \leftarrow 2^t/0.77351$;
30: return $S$;
```

It is worth mentioning that there also exist many other probabilistic counting structures, such as Loglog sketch [24] and Hyper Loglog sketch [25], but the union of these sketches cannot be easily implemented by bitwise-OR. Therefore, in our problem, we apply the FM sketch to estimate the size of the k-step expansion set, i.e., $|N_k(S)|$. The main idea of our algorithm is that we construct a FM sketch to estimate the k-step expansion ($|N_k(\{v\})|$) of each node (v). To estimate the k-step expansion of a set S ($|N_k(S)|$), we only need to do $|S| - 1$ times bitwise-OR over all the FM sketches of the nodes in S. We depict our algorithm in Alg. 2. Firstly, the algorithm calculates the personalized PageRank vector w (line 1). Secondly, the algorithm builds $|V|$ FM sketches for all nodes of the graph (line 2-11). Here we make use of the idea of the approximation neighbor function [23]. Specifically, the idea is based on the observation that the k-step expanded set of a node v_i is equivalent to the union of all the (k-1)-step expanded sets of the immediate neighbors of v_i. More formally, we have

$$N_k(\{v_i\}) = \bigcup_{(v_i,v_j)\in E} N_{k-1}(\{v_j\}). \tag{6}$$

Based on this observation, we build a FM sketch for each node v_i in a recursive manner (line 7-11). Note that we use the bitwise-OR over the FM sketches for implementing the set union operation in Eq. (6) (line 11). Finally, Alg. 2 greedily selects K nodes according to their approximate marginal gain (line 12-30). In particular, we let S be the answer set, NBP be the FM sketch representing the expanded set of the answer set S ($N_k(S)$), c be the k-step expansion of S ($|N_k(S)|$), and OBP be a temporary FM sketch representing the expanded set of $S \cup \{v_i\}$, i.e., $N_k(S \cup \{v_i\})$. Initially, Alg. 2 sets S to an empty set (line 12), NBP and OBP to 0 (line 13), and $c = 0$ (line 14). Then, Alg. 2 iteratively selects K nodes with the maximal approximate marginal gain (line 15-29). At each iteration, the algorithm chooses one node from $V - S$ (line 18-25). More specifically, for each node $v_i \in (V - S)$, Alg. 2 first estimates $|N_k(S \cup \{v_i\})|$ using the FM sketch OBP (line 19-21). Then, Alg. 2 calculates the approximate marginal gain of node v_i ($\rho_i(S) = (1 - \lambda)w_i + \frac{\lambda}{n}|N_k(S \cup \{v_i\}) - N_k(S)|$) and records the node with the maximal approximate marginal gain (line 22-25). Finally, Alg. 2 adds the node with maximal approximate marginal gain into the answer set (line 26-27) and re-estimates $|N_k(S)|$ by the FM sketch NBP (line 28-29).

Theoretically, Alg. 2 achieves $1 - 1/e - \epsilon$ approximation guarantee with high probability for the generalized diversified ranking problem, because the FM sketch approximates the k-step expansion of set S within an ϵ error bound in high probability [22]. In our experiments, we will show that the performance of Alg. 2 is desirable. In the following, we analyze the time and space complexity of Alg. 2.

Complexity analysis of the randomized greedy algorithm: The time complexity of Alg. 2 is $O(k|E|+K|V|)$. Specifically, in line 1, Alg. 2 computes the personalized PageRank vector which consumes $O(|E|)$ time complexity. In line 2-11, Alg. 2 needs to take $O(k(|E|+|V|))$ time to sketch the k-step expanded set for all nodes. In line 12-29, the algorithm takes $O(K|V|)$ time to find the answer set. Note that the bitwise-OR can be done in near constant time complexity [23]. Thus, the time complexity of Alg. 2 is $O(k|E| + K|V|)$. For the space complexity, like Alg. 1, Alg. 2 needs to store the graph G and the personalized PageRank vector w, which consumes $O(|V| + |E|)$. In addition, Alg. 2 needs to maintain $O(|V|)$ FM sketches, which takes $O(|V| \log |V|)$ bits. As a result, the space complexity of Alg. 2 is $O(|V| \log |V| + |E|)$. Notice that the space complexity of Alg. 2 is approximately $O(|E|)$, as $O(|V| \log |V|)$ can be dominated by $O(|E|)$ in most graphs. Putting it all together, we conclude that the time and space complexity of Alg. 2 is linear w.r.t. the graph size, thereby it can be scalable to large graphs.

4.3 Minimum relevance diversified measures

Besides MMR, there also exist other diversification criterions [26][27]. Here, we discuss some potential variants of the proposed diversified measures based on the minimum relevance criterion [26], where the worse-case relevance will be maximized. The minimum relevance diversified measures are given as follows:

$$J(S) = (1-\lambda)\min_{u \in S}\ w_u + \lambda\frac{|N(S)|}{n}, \tag{7}$$

and

$$J_k(S) = (1-\lambda)\min_{u \in S}\ w_u + \lambda\frac{|N_k(S)|}{n}. \tag{8}$$

Unlike $F(S)$ and $F_k(S)$, the minimum relevance diversified measures defined above are not submodular. Thus, we cannot easily design an efficient greedy algorithm with an approximation guarantee. In effect, it is easy to show that the first term of set function $J(S)$ or $J_k(S)$ is supermodular[2] [28] and the second term is submodular. Thus, the set function $J(S)$ or $J_k(S)$ is a sum over a submodular and a supermodular function, which could be approximately solved by a supermodular-submodular procedure [28]. But unfortunately, both the convergence properties and the approximation factor of the supermodular-submodular procedure are not known now. Developing efficient algorithm with performance guarantee to maximize $J(S)$ and $J_k(S)$ is an interesting future work.

5 Experiments

In this section, we evaluate the effectiveness and efficiency of the proposed approaches. Below, we first describe the experimental setup, and then report our experimental results.

5.1 Experimental setup

Datasets: We conduct our experiments on five real networks, three collaboration networks, one citation network, and one social network.

- Collaboration networks. We select three collaboration networks from Stanford network datasets [29]: namely GrQc, HepTh, and CondMat. GrQc, HepTh, and CondMat are collaboration networks collected from the e-print arXiv archive and cover all the co-authorships between authors on General Relativity and Quantum Cosmology, High Energy Physics-Theory, and Condense Matter Physics, respectively. Notice that all the collaboration networks are undirected graph.
- Citation network. We choose a citation network, namely citeHepTh, from Stanford network datasets [29]. The citeHepTh is a citation network of papers on high energy physics theory, which is originally collected from e-print arXiv archive. The citation network is a directed graph.
- The social network. Flickr is a popular photo shared website. The users in Flickr can upload photos, make friends as well as join in various interest groups. In our experiments, we employ the Flickr dataset from ASU social computing data repository [30]. The dataset contains an undirected social network with 80,513 nodes and 5,899,882 edges and 195 different groups that the users joined.

The detailed statistical information of our datasets are presented in Table 1. From Table 1, we can observe that the approximate domination number (DN) of our datasets, which is calculated by a greedy algorithm given in [20], are greater than 1,000. However, in many practical retrieval systems, users are often interested in the top-K results, where K is a small constant (eg. K=30) and it is typically smaller than the approximate DN.

TABLE 1
Summary of the datasets

name	nodes	edges	approximate DN
GrQc	5242	28,980	1,598
HepTh	9,877	51,971	2,829
CondMat	23,133	186,936	4,449
citeHepTh	27,770	352,807	3,570
Flickr	80,513	5,899,882	3,768

Evaluation metrics: In the literature, there are no well accepted measures for diversity in ranking on graphs [31]. In our experiments, we employ two metrics to measure the diversity. One is proposed in [6], which makes use of the density of the induced subgraph by the top-K ranking nodes. The density of a graph is a ratio that is equal to the number of edges existing in the graph divided by the maximum possible number of edges in the graph. Intuitively, the density inversely measures the diversity of the top-K ranking nodes. The second metric is the expansion ratio which is defined in Def. 2.1. The rationale is that the larger expansion ratio of the top-K ranking nodes indicates the better diversity. For comparing the relevance with different algorithms, we use the relevance metric given in [8]. Specifically, the relevance Rel is calculated as

$$Rel = \frac{\sum_{v_i \in S} w_i}{\sum_{v_i \in \tilde{S}} w_i}, \tag{9}$$

where S denotes the top-K diversified ranking list by the diversified ranking algorithm, $\tilde{S}$ denotes the top-K ranking list by the personalized PageRank algorithm. Note that Rel defined in Eq. (9) falls into a interval [0, 1], as the personalized PageRank algorithm always gives the K most relevant nodes. By definition, the higher Rel implies better relevant.

[2] A set function $J(S)$ is called supermodular, if $-J(S)$ is submodular.

Baselines: We compare our proposed methods with six baselines under diversity and relevance metrics defined above. For our methods, we mainly focus on k-step, for k = 1 and k = 2, denoted by **Exp**ansion-1 (Ep1) and **Exp**ansion-2 (Ep2), respectively. Ep1 and Ep2 are tested using Alg. 1 and Alg. 2, respectively. We will study the effectiveness of the parameter k in the following section. For other k-step expansions (k > 2), the performance is not significantly better than the 1-step and 2-step expansions. The six baselines are as follows.

- **P**ersonalized **P**age**R**ank (PPR): PPR is a natural competitor of our algorithm, which can be served as a baseline for evaluating relevance.
- **Gra**sshopper (Gra): Gra is a diversified ranking algorithm that leverages an absorbing random walk to achieve diversity [5]. Gra has been successfully used in diversified document summarization and ranking actors in social networks.
- **M**anifold **R**anking with **S**top **P**oints (MRSP): MRSP is proposed in [7], which is very similar to the Grasshopper algorithm. It can also be used on graphs.
- **DivR**ank (DivR): DivR makes use of the stationary distribution of a vertex reinforced random walk to rank nodes [6]. It has been applied to diversify ranking in information networks. There are two various implementation of DivR, namely pointwise DivR and cumulative DivR respectively. As reported in [6], the two algorithms achieve the similar ranking performance. Hence, we use the pointwise DivR in our experiments.
- **Dra**gon (Dra): Dra is a scalable diversified ranking algorithm [8]. Dra aims to optimize a predefined diversified ranking measure. Unlike our diversified ranking measure, the measure used in Dra lacks topological explanation, thereby it is not intuitive and reasonable to some extent.
- Diversified ranking via **R**esistive **G**raph **C**enters (RGC): RGC [10] aims to learn a diversified teleport vector to achieve diversity in ranking. However, the time complexity of RGC is cubic, thereby it cannot scale to large graphs.

We do not make comparison with the MMR algorithm [13] because [6] has shown that DivR outperforms MMR over graph datasets.

Parameter settings: In our proposed algorithms (Alg. 1 and Alg. 2), there are two common parameters: the damping factor α for computing the personalized PageRank, and the parameter λ used to tradeoff relevance and diversity. We set $\alpha = 0.85$ as it is widely used in web search. For the parameter λ, we set it to 0.5 because it is not very sensitive in our experiments. We will show the effect of λ in the following section. Additionally, for Alg. 2, we use 50 hashing functions to implement the FM sketch. For all parameters of the baseline methods, we use the same settings as given in the original papers respectively.

Experimental environment: All the experiments are conducted on a Window Server 2007 with 4xDual-Core Intel Xeon 2.66 GHz CPU, and 4G memory. All algorithms are implemented by MATLAB (R2011a).

5.2 Experimental results

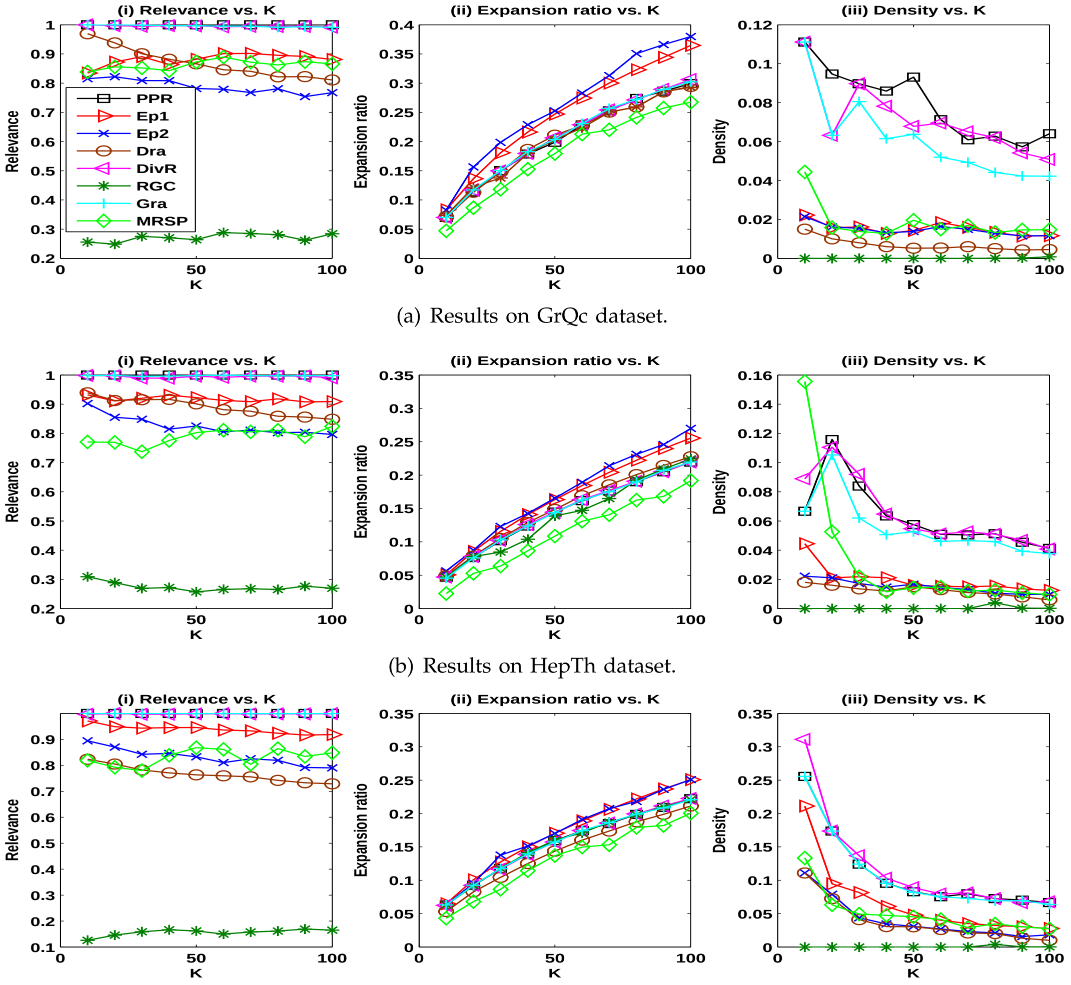
In all of our experiments, we randomly generate 100 queries, and the results are the average over all the queries. We give the detail results as follows.

Results on collaboration networks: In this experiment, we compare Ep1 and Ep2 with six baselines over three collaboration networks. Fig. 2(a), Fig. 2(b), and Fig. 2(c) depict our results on GrQc, HepTh, and CondMat datasets, respectively.

From Fig. 2(a), we can observe that DivR and Gra achieve near-optimal relevance, followed by Ep1, Dra, Ep2, MRSP, and RGC. Note that the relevance of both Ep1 and Ep2 are more than 0.8 over different K values, which indicates that our algorithms can obtain relevant results w.r.t. the queries. We can clearly see that the relevance of RGC is extremely low, which is less than 0.3 over different K values. This result implies that RGC may produce irrelevant and meaningless results. For the diversity, we find that Ep2 is the winner under the expansion ratio metric among all the algorithms. Besides, Ep1 also outperforms other baselines under the expansion ratio metric. The expansion ratio by DivR, Gra, and MRSP are slightly worse than PPR, which suggests that DivR, Gra, and MRSP do not perform well to enhance diversity in collaboration networks under the expansion ratio metric. Under the density metric, RGC outperforms the competitors (recall that smaller density implies better diversity). Ep1, Ep2, and MRSP achieve comparable density, and they are slightly worse than Dra. DivR and Gra also do not perform well under the density metric. Similar results can be observed in HepTh and CondMat datasets.

Based on the observations, on the collaboration networks, we conclude that DivR, Gra, and MRSP do not perform well regarding diversity. The reason would be that these algorithms lack a clear explanation for diversity. RGC exhibits excellent performance for improving diversity, but it significantly sacrifices the performance of relevance. Our Ep1 and Ep2 as well as Dra achieve a good tradeoff between the relevance and the diversity. The reason is that our algorithms and Dra have a clear objective to optimize the predefined diversified ranking measures. Moreover, our algorithms exhibit better relevance and better expansion ratio than Dra.

Results on citation network: Unlike the collaboration network, the citation network is a directed graph. Here, we test MRSP by ignoring the direction of the edges as MRSP cannot be directly applied to the directed graphs. Fig. 3 describes our results.



(a) Results on GrQc dataset.

(b) Results on HepTh dataset.

(c) Results on CondMat dataset.

Fig. 2. Comparison of various diversified ranking algorithms on collaboration networks (color online).

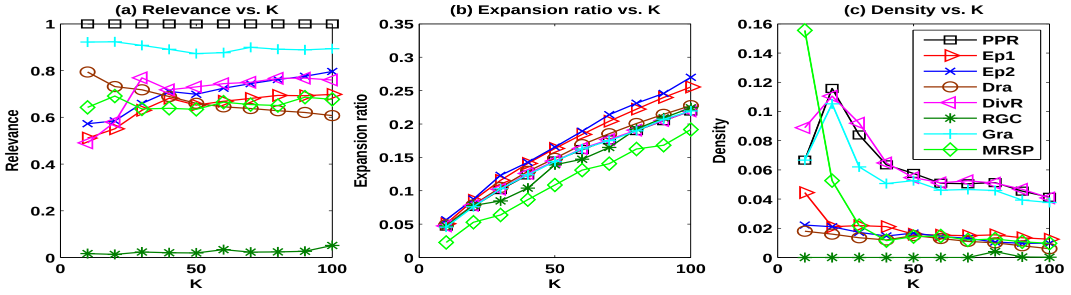


Fig. 3. Comparison of various diversified ranking algorithms in citeHepTh dataset.

From Fig. 3, we find that Gra outperforms other algorithms by relevance metric. RGC shows the lowest relevance, which suggests that RGC may generate completely irrelevant ranking results. For other baselines except PPR, they show comparable relevance. For our approaches, Ep1 shows better relevance than Ep2. For the diversity, Ep2 outperforms the other algorithms under the expansion ratio metric. The expansion ratio by Ep1 is better than the expansion ratio by the six baseline algorithms. However, under the density metric, we can observe that RGC gets the best performance. Our approaches, MRSP, and Dra achieve comparable density. Also, for our approaches, Ep2 is slightly better than Ep1 under the density metric. In general, the results on the citation networks consist with the results on the collaboration networks.

Results on Flickr social network: Here we test our proposed algorithms in Flickr social network. Our goal is to find the top-K users who not only have higher personalized PageRank scores relative to the queries, but also cover as many interest groups as possible. Hence, in addition to the diversity measures described in Section 5.1, we introduce the "group

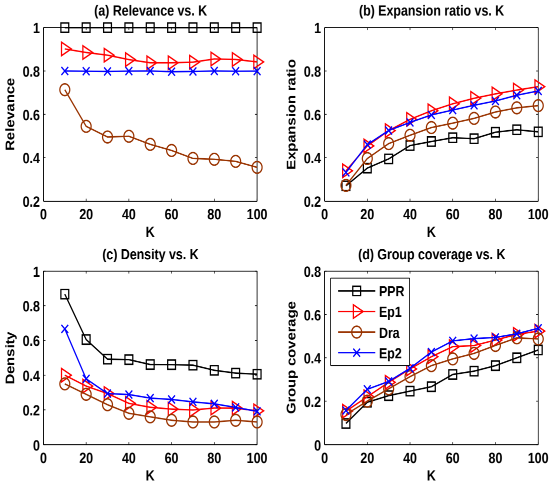


Fig. 4. Comparison of various diversified ranking algorithms in Flickr social network.

coverage" as a new diversity measure in this experiment. Intuitively, the more groups that are covered by the top-K ranking list the better diversity it has. In this experiment, we only compare our Ep1 and Ep2 with PPR and Dra. The reason is of twofold. First, the other baselines either cannot get answers in 12 hours or cannot be conducted due to their memory requirements. Second, as observed in our previous experiments, Dra outperforms the other baselines. Our results are shown in Fig. 4. From Fig. 4, we can observe that both Ep1 and Ep2 significantly outperform Dra based on the relevance, the expansion ratio, and the group coverage metrics. More specifically, under the relevance and expansion ratio metrics, Ep1 is clearly the best performer among all the diversified ranking algorithms. Also, notice that the relevance by Dra decreases as the K increases. When K=100, Dra exhibits low relevance (less than 0.4). Instead, our algorithms show quite robust relevance w.r.t. different K values. Furthermore, the relevance of our algorithms are greater than 0.8 over various K values. Under the density metric, Dra slightly outperforms Ep1 and Ep2. However, under the group coverage metric, Ep2 achieves the best performance, followed by the Ep1, Dra, and then PPR. From the practical point of view, the performance of our algorithms are better than the performance of Dra, because the ranking results by our algorithms cover more interest groups than that of Dra. The reason can be that our diversified ranking measures capture the topological properties of the graph, which is more intuitive and reasonable than the measure used in Dra.

To summarize, over all of our experiments, we make the following observations. (1) DivR and Gra achieve near-optimal relevance but their performance of improving diversity is quite low. (2) RGC gets near-optimal diversity under the density metric, but it exhibits extremely low relevance. (3) The performance of MRSP is very low under the expansion ratio metric (even worse than PPR). (4) Ep1, Ep2, and Dra show a good balance between the relevance and the diversity. Moreover, our Ep1 and Ep2 exhibit better relevance and diversity than Dra over most datasets used.

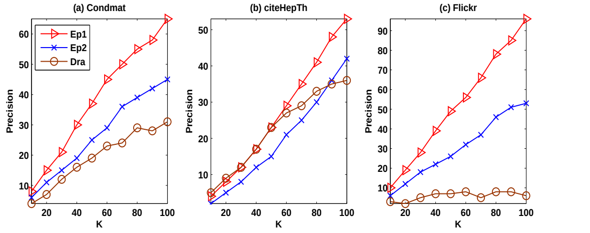


Fig. 5. Comparison of precision of Ep1, Ep2, and Dra.

Precision comparison: To further evaluate the effectiveness of our algorithms, we compare the precision of our approaches with the state-of-the-art Dra. Since there is no ground truth in graph-type datasets, we use the personalized PageRank as the ground-truth rank which is also used in [8]. The precision is defined by the following formula:

$$Pre = |S \cap \tilde{S}|/|\tilde{S}|, \qquad (10)$$

where S and $\tilde{S}$ is defined in Eq. (9). Fig. 5 depicts our results in Condmat, citeHepTh, and Flickr datasets. Similar results can be observed in other datasets. From Fig. 5, we can clearly see that both Ep1 and Ep2 consistently outperform Dra in Condmat and Flickr datasets over different K. In citeHepTh dataset, we can observe that all three algorithms generate comparable rank, and the performance of Ep1 is slightly better than Dra. The performance of Dra is not very stable over our datasets. In citeHepTh dataset, the performance of Dra is comparable to our algorithms, but in in Flickr dataset, Dra does not perform well (precision is lower than 10 given K = 100). This result implies that Dra produces less meaningful rank in Flickr dataset. In contrast to Dra, the performance of our algorithms is very stable over different datasets. In this sense, we can conclude that our algorithms are better than Dra.

Time comparison: We compare the average query processing time of various diversified ranking algorithms over five network datasets. We take the average on the query processing time of the ranking algorithms over different K values and different queries. Table 2 shows our results. From Table 2, we can observe that PPR is the most efficient algorithm. Ep1 and Dra achieve competitive efficiency with PPR. Ep2 is slightly worse than Ep1, Dra, and PPR, but is still very efficient due to the linear time and space complexity. For the other baselines, we can clearly see that their time requirements are very high. More worse, on the Flickr dataset, RGC, Gra, and MRSP cannot get the top-K ranking results in 12 hours, and DivR cannot be conducted due to its memory

TABLE 2
Average query time of various algorithms (in second).

	GrQc	HepTh	CondMat	citeHepTh	Flickr
PPR	0.02	0.08	0.26	0.37	4.96
EP1	0.03	0.10	0.31	0.56	9.82
EP2	0.07	0.20	1.18	2.33	18.32
Dra	0.03	0.10	0.30	0.53	8.74
DivR	51.03	186.04	1099.09	1600.59	–
RGC	185.55	928.12	7825.72	8446.38	–
Gra	90.04	241.30	879.04	1277.35	–
MRSP	90.17	231.86	887.92	1601.78	–

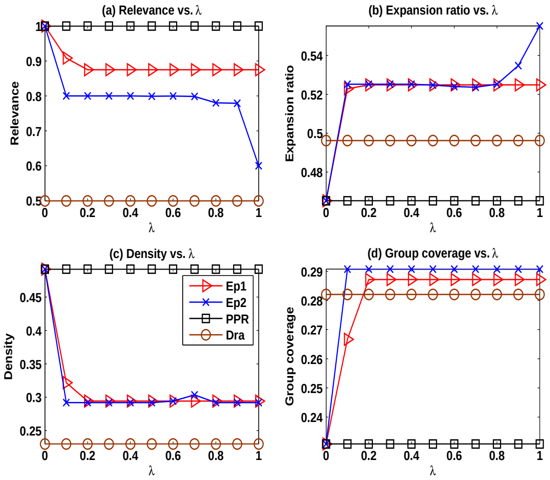


Fig. 6. The effect of parameter λ.

requirement. This results confirm our time and space complexity analysis in the previous sections.

Effect of parameter λ: We study the effect of the parameter λ in Ep1 and Ep2, i.e. λ in Eq. (2) and Eq. (5), which is leveraged to tradeoff the relevance and the diversity. Here we study the top 30 ranking results (K=30) under different λ values in Flickr dataset. Similar results can be observed in other datasets and for other K. We use the results of PPR and Dra as the baselines. The reasons are (1) the ranking result by PPR is a natural measure for relevance, and (2) Dra outperforms other baselines. The results are depicted in Fig. 6. As can be seen in Fig. 6(a), the relevance by Ep2 decreases as λ increases, while the relevance by Ep1 is robust w.r.t. λ. For the relevance, both Ep1 and Ep2 outperform Dra. According to Fig. 6(b), Fig. 6(c), and Fig. 6(d), we can observe that the diversity by Ep1, which is measured by the expansion ratio, density, and group coverage, generally increases as λ increases. This is because a larger λ means more weights are assigned, in order to improve the diversity in our diversified measure (Eq. (2)). We also find that Ep1 is very robust w.r.t. λ. In addition, we can clearly see that both Ep1 and Ep2 outperform Dra by the expansion ratio and group coverage measures, while by density measure, our algorithms are slightly worse than Dra.

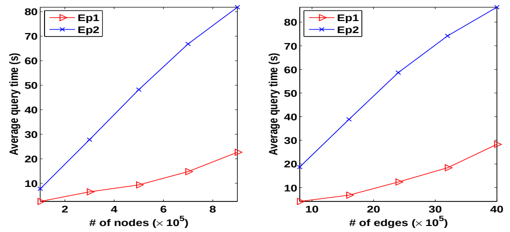


Fig. 7. Scalability of the proposed algorithms.

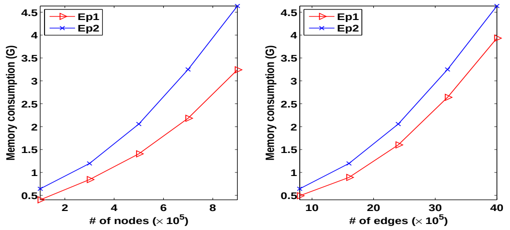


Fig. 8. Memory consumption of the proposed algorithms.

Scalability testing and memory consumption: To study the scalability of Ep1 and Ep2, we generate two sets of synthetic graphs G with nodes ranging from 100,000 to 900,000 and edges from 800,000 to 4,000,000 using the Erdos-Renyi random graph model, respectively. Here we set K = 30, and similar results can be observed for other K. Our results are described in Fig. 7. From Fig. 7, we can clearly see that both Ep1 and Ep2 scale linearly w.r.t. both the numbers of nodes (left part of Fig. 7) and edges (right part of Fig. 7). Therefore, our Ep1 and Ep2 can be used for very large graphs. The results confirm our time complexity analysis in the previous sections.

To validate the space complexity of our algorithms, in Fig. 8, we show the memory consumption of our algorithms in the same set of synthetic graphs. Specifically, in the left part of Fig. 8, we can see that the memory consumption of both Ep1 and Ep2 increase as the number of nodes increases. The curves of both Ep1 and Ep2 become a line when the number of nodes is larger than 500,000. Similarly, from the right part of Fig. 8, we can observe that the memory consumption of both Ep1 and Ep2 increase as the number of edges increases, and the curves of Ep1 and Ep2 tend to be a line when the number of edges is larger than 2,400,000. These results confirm the linear space complexity of our algorithms.

Performance of Alg. 2: It is worth noting that Alg. 2 gives an approximate answer instead of the exact answer given by Alg. 1. We evaluate the approximation performance of Alg. 2. To this end, firstly, we use Alg. 2 to test the 1-step expansion (set k=1 in Alg. 2), and we refer to it as Approx. Ep1. We compare the performance of Approx. Ep1 with Ep1, which is implemented by Alg. 1. Fig. 9 shows our results in Flickr dataset. Similar results can be observed in other datasets. From Fig. 9(a), we can find that Approx. Ep1 shows better relevance than Ep1. However, from Fig. 9(b), (c), and (d), Approx. Ep1 is slightly worse than Ep1 under the three diversity

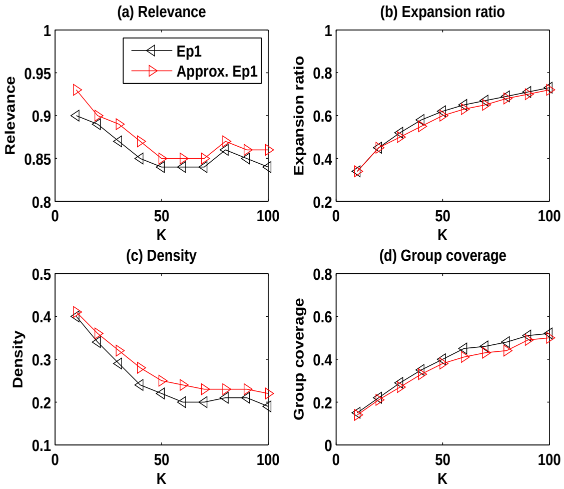


Fig. 9. Performance of the randomized greedy algorithm.

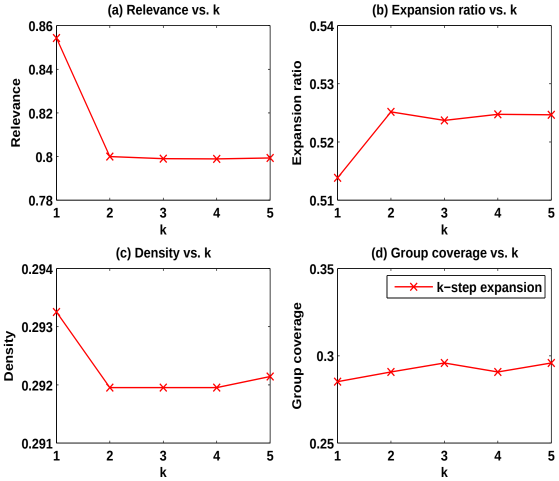


Fig. 10. The effect of parameter k in k-step expansion based algorithms.

metrics. Overall, Approx. Ep1 achieves comparable performance with Ep1. This results suggest that our randomized greedy algorithm (Alg. 2) can achieve a good performance guarantee, which consists with our analysis in Section 4.

Effect of parameter k: We investigate how the parameter k affects the performance of the k-step expansion based algorithms, which are implemented by Alg. 2. Fig. 10 shows our results in Flickr dataset, and the similar results can be observed in other datasets. From Fig. 10, we can see that the relevance and diversity are generally not sensitive w.r.t. different k when k $\geq$ 2. The 2-step expansion (k=2) achieves the best expansion ratio and density, thereby in our previous experiments we set k=2.

6 Related Work

Diversified ranking on text data: Diversity has been recognized as important criteria in information retrieval. There are a large body of works on query or search results diversification [13][32][33][34][35][36][37]. In document retrieval, one of a well-known method is the maximal marginal relevance (MMR) proposed by Carbonell and Goldstein [13], which achieves diversity by maximizing a linear combination function that captures both dissimilarity among the results and relevance w.r.t. the query. After Carbonell and Goldstein's work, many approaches addressing diversification have been proposed in recent years. Zhai, et al. [38] propose a subtopic retrieval approach to results diversification. Agrawal, et al. [39] formulate the query results diversification as a submodular function maximization problem. Gollapudi, et al. [26] present several axioms for query results diversification. All the above mentioned methods primarily address to documents data. An excellent survey on query results diversification is given in [27].

Submodular set function maximization: Our diversified ranking problem is closely related to submodular set function maximization problem, which is generally NP-hard. However, there always exists a near-optimal greedy algorithm for solving such problem [17]. There are many applications that have been formulated as a submodular set function maximization problem such as influence maximization problem in social networks [40], observation selection and sensor placement problem [41], [42], document summarization problem [43], [37], as well as the set cover problem [44]. In this paper, we formulate the diversified ranking problem on graphs as the submodular set function maximization problem.

Expansion on graphs: Our work is also related to the expansion of a graph, which is a well known concept in expander graph theory [12]. This concept recently is used for sampling community structure [45] and facilitating decentralized search in networks [14]. However, our definition of expansion is different from the previous work, and we leverage expansion to measure diversity of the top-K ranking results.

7 Conclusions

In this paper, we present a study of finding top-K diversified ranking on graphs. Firstly, we propose a novel diversified ranking measure, which captures both relevance and diversity. Secondly, we prove the submodularity of this measure and design an efficient greedy algorithm to achieve near-optimal diversified ranking. The proposed method has linear time and space complexity w.r.t. the size of the graph, thus it can be scalable to large graphs. Thirdly, we present a generalized diversified ranking measures and develop an efficient randomized greedy algorithm for maximizing it accurately. Finally, extensive experiments show the effectiveness, efficiency and scalability of the proposed methods.

Acknowledgments

The work was supported by grant of the Research Grants Council of the Hong Kong SAR, China No. CUHK/419109.

References

[1] S. Brin and L. Page, "Pagerank: Bringing order to the web," Stanford Digital Library Project, Tech. Rep., 1997.

[2] M. E. J. Newman, *Networks: An Introduction*. OXFORD University Press, 2010.

[3] T. H. Haveliwala, "Topic-sensitive pagerank," in *WWW '02*.

[4] G. Jeh and J. Widom, "Scaling personalized web search," in *WWW '03*.

[5] X. Zhu, A. B. Goldberg, J. V. Gael, and D. Andrzejewski, "Improving diversity in ranking using absorbing random walks," in *HLT-NAACL'07*.

[6] Q. Mei, J. Guo, and D. R. Radev, "Divrank: the interplay of prestige and diversity in information networks," in *KDD '10*.

[7] X. Zhu, J. Guo, X. Cheng, P. Du, and H. Shen, "A unified framework for recommending diverse and relevant queries," in *WWW '11*.

[8] H. Tong, J. He, Z. Wen, R. Konuru, and C.-Y. Lin, "Diversified ranking on large graphs: an optimization viewpoint," in *KDD*, 2011.

[9] R.-H. Li and J. X. Yu, "Scalable diversified ranking on large graphs," in *ICDM*, 2011, pp. 1152–1157.

[10] A. Dubey, S. Chakrabarti, and C. Bhattacharyya, "Diversity in ranking via resistive graph centers," in *KDD*, 2011, pp. 78–86.

[11] O. Haggstrom, *Finite markov chains and algorithmic applications*. Cambridge University Press, 2002.

[12] S. Hoory, N. Linial, and A. Wigderson., "Expander graphs and their applications," *Bull. Amer. Math. Soc.*, vol. 43, pp. 439–561, 2006.

[13] J. G. Carbonell and J. Goldstein, "The use of mmr, diversity-based reranking for reordering documents and producing summaries," in *SIGIR '98*.

[14] A. S. Maiya and T. Y. Berger-Wolf, "Expansion and search in networks," in *CIKM '10*.

[15] R. Kumar, K. Punera, and A. Tomkins, "Hierarchical topic segmentation of websites," in *KDD' 06*.

[16] T. Lappas, E. Terzi, D. Gunopulos, and H. Mannila, "Finding effectors in social networks," in *KDD' 10*.

[17] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions-i," *Mathematical Programming*, vol. 14, pp. 265–294, 1978.

[18] U. Feige, "A threshold of ln n for approximating set cover," *J. ACM*, vol. 45, pp. 634–652, 1998.

[19] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. S. Glance, "Cost-effective outbreak detection in networks," in *KDD*, 2007.

[20] T. W. Haynes, S. T. Hedetniemi, and P. J. Slater, *Domination in graphs: advanced topics*. MARCEL DEKKER, INC, 1998.

[21] T. H. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms, Third Edition*. MIT Press, 2011.

[22] P. Flajolet and G. N. Martin, "Probabilistic counting algorithms for data base applications," *J. Comput. Syst. Sci.*, vol. 31, no. 2, pp. 182–209, 1985.

[23] C. R. Palmer, P. B. Gibbons, and C. Faloutsos, "Anf: a fast and scalable tool for data mining in massive graphs," in *KDD*, 2002, pp. 81–90.

[24] M. Durand and P. Flajolet, "Loglog counting of large cardinalities (extended abstract)," in *ESA*, 2003, pp. 605–617.

[25] P. Flajolet, E. Fusy, O. Gandouet, and F. Meunier, "Hyperloglog: the analysis of a near-optimal cardinality estimation algorithm," in *ESA*, 2003, pp. 605–617.

[26] S. Gollapudi and A. Sharma, "An axiomatic approach for result diversification," in *WWW '09*.

[27] M. Drosou and E. Pitoura, "Search result diversification," *SIGMOD Rec.*, vol. 39, pp. 41–47, 2010.

[28] N. Narasimhan and J. Bilmes, "A supermodular-submodular procedure with applications to discriminative structure learning," in *UAI'05*.

[29] J. Leskovec, "Standford network analysis project," 2010. [Online]. Available: http://snap.standford.edu

[30] R. Zafarani and H. Liu, "Social computing data repository at ASU," 2009. [Online]. Available: http://socialcomputing.asu.edu

[31] F. Radlinski, P. N. Bennett, B. Carterette, and T. Joachims, "Redundancy, diversity and interdependent document relevance," *SIGIR Forum*, vol. 43, 2009.

[32] Y. Zhang, J. P. Callan, and T. P. Minka, "Novelty and redundancy detection in adaptive filtering," in *SIGIR '02*.

[33] C.-N. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen, "Improving recommendation lists through topic diversification," in *WWW '05*.

[34] C. L. A. Clarke, M. Kolla, G. V. Cormack, O. Vechtomova, A. Ashkan, S. Büttcher, and I. MacKinnon, "Novelty and diversity in information retrieval evaluation," in *SIGIR '08*.

[35] H. Ma, M. R. Lyu, and I. King, "Diversifying query suggestion results," in *AAAI'10*.

[36] E. Minack, W. Siberski, and W. Nejdl, "Incremental diversification for very large sets: a streaming-based approach," in *SIGIR*, 2011, pp. 585–594.

[37] H. Lin and J. Bilmes, "A class of submodular functions for document summarization," in *ACL*, 2011, pp. 510–520.

[38] C. Zhai, W. W. Cohen, and J. D. Lafferty, "Beyond independent relevance: methods and evaluation metrics for subtopic retrieval," in *SIGIR '03*.

[39] R. Agrawal, S. Gollapudi, A. Halverson, and S. Ieong, "Diversifying search results," in *WSDM '09*.

[40] D. Kempe, J. M. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *KDD*, 2003, pp. 137–146.

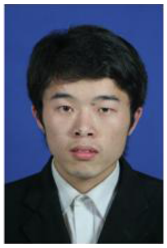
[41] A. Krause and C. Guestrin, "Near-optimal observation selection using submodular functions," in *AAAI*, 2007, pp. 1650–1654.

[42] A. Krause, A. P. Singh, and C. Guestrin, "Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies," *Journal of Machine Learning Research*, vol. 9, pp. 235–284, 2008.

[43] H. Lin and J. Bilmes, "Multi-document summarization via budgeted maximization of submodular functions," in *HLT-NAACL*, 2010.

[44] V. V. Vazirani, *Approximation Algorithms*. Springer, 2004.

[45] A. S. Maiya and T. Y. Berger-Wolf, "Sampling community structure," in *WWW '10*.



Rong-Hua Li Rong-Hua Li is pursuing his PhD degree in Department of System Engineering and Engineering Management, The Chinese University of Hong Kong, Hong Kong. His research interests include social network analysis and mining, complex network theory, uncertain graphs mining, Monte-Carlo algorithms, and machine learning.

Jeffery Xu Yu Jeffrey Xu Yu received the BE, ME, and the PhD degrees in computer science from the University of Tsukuba, Japan, in 1985, 1987, and 1990, respectively. He held teaching positions in the Institute of Information Sciences and Electronics, University of Tsukuba, Japan, and the Department of Computer Science, The Australian National University. Currently, he is a professor in the Department of Systems Engineering and Engineering Management, the Chinese University of Hong Kong. He is serving as a VLDB Journal editorial board member. His current main research interest includes graph database, graph mining, keyword search in relational databases, and social network analysis.