Information Processing and Management xxx (xxxx) xxx–xxx

Contents lists available at ScienceDirect

Information Processing and Management

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

Search result diversification on attributed networks via nonnegative matrix factorization

Zaiqiao Meng[a,*], Hong Shen[*,a,b], Huimin Huang[a], Wei Liu[a], Jing Wang[c], Arun Kumar Sangaiah[d]

[a] School of Data and Computer Science, Sun Yat-sen University, China
[b] School of Computer Science, The University of Adelaide, Adelaide, Australia
[c] Neusoft Institute, Guangdong, China
[d] School of Computing Science and Engineering, VIT University, Vellore, India

ARTICLE INFO

Keywords:
Graph search
Diversification
Nonnegative matrix factorization
Attributed network

ABSTRACT

Search result diversification is an effective way to tackle query ambiguity and enhance result novelty. In the context of large information networks, diversifying search result is also critical for further design of applications such as link prediction and citation recommendation. In previous work, this problem has mainly been tackled in a way of implicit query intent. To further enhance the performance on attributed networks, we propose a novel search result diversification approach via nonnegative matrix factorization. Our approach encodes latent query intents as well as nodes as representation vectors by a novel nonnegative matrix factorization model, and the diversity of the results accounts for the query relevance and the novelty w.r.t. these vectors. To learn the representation vectors of nodes, we derive the multiplicative updating rules to train the nonnegative matrix factorization model. We perform a comprehensive evaluation on our approach with various baselines. The results show the effectiveness of our proposed solution, and verify that attributes do help improve diversification performance.

1. Introduction

Social network platforms such as Facebook provide a wealth of information resources that enable users to search information about their opinions, status and friends. To effectively search which users are similar to a query user in an information network is critical for further design of applications such as community detection (Cheng, Zhou, & Yu, 2011), social recommendation (Lee & Brusilovsky, 2017) and link prediction (Wu, Zhang, & Ren, 2017). As an inherently limited representation of a complex information need, queries submitted to information networks system is often ambiguous to some extent. While real social networks keep expanding in scales, the search result and recommended list that can be presented to users is only in a limited number. Hence, reranking the search results in order to meet the ambiguous complex information needs is challenging.

Search result diversification as an effective way to tackle query ambiguity and enhance result novelty has gained much attention in various fields, including text retrieval (Liang, Ren, & De Rijke, 2014a; Liang & de Rijke, 2015), recommender systems (Küçüktunç, Saule, Kaya, & Çatalyürek, 2015), expert finding (Liang & de Rijke, 2016) and graph based ranking (Li & Yu, 2013). A number of algorithms of search result diversification have been proposed, which can be simply classified into two categories according to whether the *query aspects*, the aspects of information needs behind a query, is implicit or explicit. For those implicit approaches,

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

https://doi.org/10.1016/j.ipm.2018.05.005
Received 20 January 2018; Received in revised form 23 March 2018; Accepted 15 May 2018

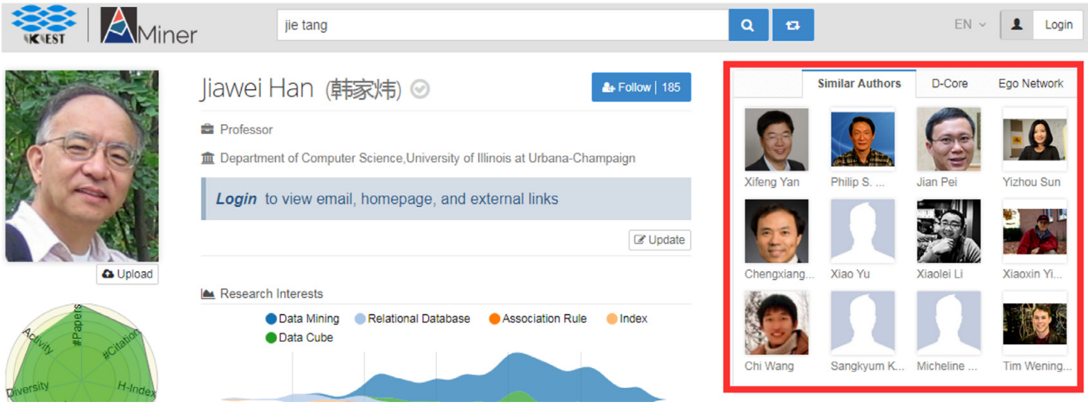


Fig. 1. An example of recommendation in AMiner academic social network.

clearly defined intent of query might be underspecified or implicit. It is not clear which aspects behind this query is what users actually interested in. To diversify the search results under this problem setting, the idea is to make the search results covering certain diversity goodness as much as possible so that the majority users with different backgrounds could find the results to be relevant to their needs (Agrawal, Gollapudi, Halverson, & Ieong, 2009; Li & Yu, 2013; Liang, Cai, Ren, & de Rijke, 2016; Liang, Yilmaz, Shen, de Rijke, & Croft, 2017). For those explicit approaches, a set of aspects underlying the query intent might be explicitly available. Rather than aiming to satisfy as many users as possible, the problem becomes to select a set of representative candidates that collectively provide maximum relevance w.r.t. these explicit query aspects (Liang, Ren, & De Rijke, 2014b). For the explicit query aspect diversification problem, quite a lot of past work has focused on explicitly modeling the possible information needs underlying a query. It has shown that explicit approaches are usually somewhat superior to implicit diversification techniques in text retrieval (Santos, Macdonald, & Ounis, 2010).

Diversifying the searching and ranking results on large information networks has also become an increasingly crucial problem, as it has a large number of potential motivating applications such as link prediction (Zhang, Yu, & Zhou, 2014), citation recommendation (Küçüktunç et al., 2015), as well as personalized services for social networks (Li & Yu, 2013).

For instance, in Fig. 1, we show a non-diversified solution of author recommendation for Professor Jiawei Han in AMiner.[1] AMiner provides a similar authors list to let users more easily find their interested authors when they visit one's personal page. However, the recommended list of this non-diversified strategy lacks of novelty. Since Professor Han's research interests are with many areas, and if only considering the similarity, the recommended result will only present one of his main research areas. As a result, users may lose their interest when they go the first few recommendations over. In this situation, making the limited sized recommended list diverse should be considered.

Existing diversification approaches in searching of networks are typically built on implicit representation of the query intent, based upon some global diversified goodness of nodes, such as rich-gets-richer mechanism (Mei, Guo, & Radev, 2010), the direction-aware goodness (Tong, He, Wen, Konuru, & Lin, 2011) and the neighbor expansion (Li & Yu, 2013). However, it is questionable that nodes with such global diversified goodness are what query users are actually interested in. These works try to identify the 'correct' interpretation of diversified intent behind any query, without taking into account the intent of the query node itself. Consequently, important aspects of the query node may be overlooked simply because they are not well represented among the whole network; conversely, marginally important aspects may be overemphasized due to the heterogeneity of networks. Moreover, nodes in real-world information networks, such as online social networks and scientific collaboration networks, are often associated with a rich set of features or attributes. Existing diversified search algorithms focus on pure networks only and ignore that the attributes also present diverse features.

Therefore, in this paper, we study the problem of *search result diversification on attributed networks*. In particular, given an information network $\mathbf{G}$ with node attributes matrix $\mathbf{A}$, a query node q and result size k, our problem is searching for a result set of k nodes that are similar to q and diverse to each other w.r.t. the query aspects. In this problem settings, the query aspects can be represented by the local subgraphs that contain the query node, e.g. research interest groups in AMiner and social circles in Facebook.

To address this problem, we propose a new search result diversification approach called **NMFDIV**, in which query aspects and nodes are encoded as representation vectors and these representation vectors then can be directly leveraged for diversifying result nodes. More precisely, our NMFDIV model first learns low-dimensional feature representation vectors of nodes through a nonnegative matrix factorization model, then select representative nodes according to a combined diversification optimization of aspect relevance and result novelty.

Our contributions can be summarized as follows:

- We tackle the challenges of search result diversification on attributed networks in a novel way that represents the possible

[1] http://www.aminer.org/. An academic social network platform.

information needs underlying a query as a representation vector, and the diversity of the result accounts for the overall coverage of each retrieved nodes and the novelty among these nodes w.r.t. their representation vectors.

- We propose a novel representation learning model that models edges, attributes and similarities to the query node by the non-negative matrix factorization method, and derive efficient multiplicative updating rules to learn this model efficiently.
- We conduct diversified search experiments on several real-world datasets to verify the efficiency and effectiveness of our method, and show that our approach significantly outperforms the baselines on S-recall metric.

The rest of the paper is organized as follows. Section 2 briefly surveys the related work. Section 3 formally states the problem of this paper. In Section 4 we detail our method to address the problem. Section 5 reviews the results of our experiments. Finally, we conclude the paper in Section 6.

2. Related work

The work presented in this paper is closely related to *search result diversification on networks, search result diversification on documents*, and *nonnegative matrix factorization*.

2.1. Search result diversification on networks

There are several studies on search result diversification on network data. DivRank (Mei et al., 2010) employs a time-variant random walks process to facilitate the rich-gets-richer mechanism in node ranking. Tong et al. (2011) propose a scalable diversified ranking algorithm by optimizing a predefined direction-aware diversified goodness measure. Recently a neighbor expansion based diversified ranking method has been proposed, with the assumption that nodes with large neighbor expansion would be dissimilar to each other (Li & Yu, 2013). Küçüktunç, Saule, Kaya, and Çatalyürek (2013) propose 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. However, all of these works are based on the premise that intents of the query are implicit. Instead, we tackle the problem in an explicit way that we infer vector representation for each node underlying the aspects of query node.

2.2. Search result diversification on documents

Many search result diversification models have been proposed in the field of information retrieval, which can be classified as either implicit or explicit (Santos et al., 2010; 2011). Representative implicit methods include the MMR model (Carbonell & Goldstein, 1998), probabilistic model (Chen & Karger, 2006) and subtopic retrieval model (Zhai, Cohen, & Lafferty, 2003). Well-known explicit examples include xQuAD (Santos et al., 2010), IA-select (Agrawal et al., 2009), fusion-based diversification (Liang et al., 2014a), streaming diversification (Liang et al., 2017) and PM-2 (Dang & Croft, 2012). Instead of implicitly aspects assumption, these algorithms explicitly model aspects of the query according to a taxonomy (Agrawal et al., 2009), click logs and anchor text (He, Hollink, & de Vries, 2012) or topic model (Liang et al., 2014a). The diversification algorithms proposed by Liang et al. (2014a,b); Liang et al. (2017) are the most effective diversification algorithms so far, which serves for diversifying documents given a query. However, they diversifying the search result on documents, rather than networks.

2.3. Nonnegative matrix factorization

NMF (Nonnegative Matrix Factorization) (Lee & Seung, 2001; Wang & Zhang, 2013-06) is a popular machine learning method initially used to finding parts-based and linear representations of nonnegative data. It finds two non-negative matrices whose product provides a good approximation to the input matrix. Existing NMF algorithms can be divided into four categories: Basic NMF (Seung & Lee, 1999), Constrained NMF (Deng Cai, Xiaofei He, Jiawei Han, & Huang, 2011-08), Structured NMF (Chakraborty & Sycara, 2015), and Bayesian NMF (Psorakis, Roberts, Ebden, & Sheldon, 2011) according to the design principles, problems, relationships and evolution of these algorithms. The advantages of this parts-based representation have been observed in many real-world problems such as face analysis (Hoyer, 2004), document clustering (Xu, Liu, & Gong, 2003), and social network analysis (Chakraborty & Sycara, 2015). Recently, the matrix factorization techniques have also been applied to attribute network representation learning (Huang, Li, & Hu, 2017a; 2017b). Unlike these network representation learning models, our model is to learn node representation to solve the search result diversification tasks.

3. Problem statement

In this paper, scalars are denoted by lowercase alphabets (e.g., n). Vectors are represented by boldface lowercase alphabets (e.g., $\mathbf{u}$). Matrices are denoted by boldface uppercase alphabets (e.g., $\mathbf{U}$). The ith row of a matrix $\mathbf{U}$ is denoted by $\mathbf{u_i}$. The (i, j)th element of a matrix is denoted by $\mathbf{u_{ij}}$. The transpose of $\mathbf{U}$ is represented as $\mathbf{U}^{\mathrm{T}}$. The dot product of two vectors is denoted by $\mathbf{a} \cdot \mathbf{b}$. The $l1$-norm of a vector is denoted by $\|\cdot\|_1$. The Frobenius norm of a matrix is represented as $||\cdot||_F$.

Let $\mathscr{G} = (\mathbf{G}, \mathbf{A})$ be an undirected attributed network with n interconnected nodes, where $\mathbf{G}$ is a $n \times n$ real-valued adjacency symmetric matrix with the edge weight $\mathbf{g}_{ij}$ being its elements. Let $\mathbf{A}$ be a $n \times m$ matrix that collects all of the node attributes, where m is the number of node attribute categories, and element $\mathbf{a}_{ij}$ describes attribute j associated with node i. We also use $\mathscr{V}$ to represent the node set of $\mathscr{G}$.

Based on the terminologies explained above, we formally define the problem of ***Search Result Diversification on Attributed Networks*** **(SRDAN)** as follows:

Given an attributed network **G**, *a query node q, result size k, the SRDAN problem is to search for a result set $\mathscr{S}$ with k nodes, such that: (1) the possible aspects of q can be quantitatively measured, (2) all the nodes in $\mathscr{S}$ have the maximum relevance w.r.t. aspects underlying q, and (3) all the nodes are novel to each other.*

4. Our method

4.1. Overview of our algorithm

To solve the SRDAN problem, we propose a new search result diversification approach called NMFDIV, which consists of two main steps: (I) represent all the interest aspects of q; (II) compute the relevance of nodes to each aspect and performs diversification to ensure the novelty.

In the first step, explicit diversification approaches in the field of text retrieval directly model query aspects according to the query log of a commercial search engine (Santos, Macdonald, & Ounis, 2011) or document topic (Liang et al., 2014a), actively seeking to maximize the coverage of their selected documents w.r.t. these aspects. In order to quantitatively measure the various query aspects on attributed networks, we propose a novel NMF model that represents all nodes of a network in a unified d-dimensional vector space while preserving the node proximity both in topological structure **G** and node attributes **A**. Additionally, our NMF model ensures that all the important aspects of query can be well expressed in its d-dimensional vector space. We will detail this NMF model in the next subsection.

In the second step, our method uses the outputs of first step as input into an existing diversification method. There are many diversification approaches available for diversifying the search results, including MMR (Carbonell & Goldstein, 1998), MVA (Wang & Zhu, 2009), and MaxSum (Gollapudi & Sharma, 2009). In this paper, we use the MaxSum because it is a state-of-the-art search result diversification algorithm (Borodin, Lee, & Ye, 2012; Gollapudi & Sharma, 2009). Given a fixed integer k, the goal of MaxSum is to find a subset $\mathscr{S} \in \mathscr{V}$ that:

$$\begin{aligned} \max_{\mathscr{S}\in\mathscr{V}} \quad & f_{\text{MaxSum}}(\mathscr{S}) = \sum_{u\in\mathscr{S}} rel(u, q) - \theta \sum_{u,v\in\mathscr{S}} rel(u, v), \\ s.\,t. \quad & |\mathscr{S}| = k, \end{aligned} \tag{1}$$

where $\theta > 0$ is a free parameter. In the above objective function, the first term on the right side measures the relevance between the query node and the result nodes, and the second term makes the dissimilarity between two result nodes as large as possible, where $rel(\cdot,\ \cdot)$ is the relevance function, and θ governs the trade-off between these two terms. MaxSum maximizes the relevance of result nodes to query aspects while ensuring the novelty. Thus, the condition (2) and (3) of SRDAN can be achieved by using this optimization function. Here we simply use the Euclidean metric to measure the relevance of vectors between two nodes. Although maximizing the objective of Eq. (1) is NP-hard, there is a simple linear time (w.r.t. n and k) greedy algorithm that can obtain a 2-approximation solution (Borodin et al., 2012; Gollapudi & Sharma, 2009). The pseudocode of NMFDIV is presented in Algorithm 1.

4.2. Attributed network representation learning through NMF

We denote the representation of nodes as matrix **U** with $\mathbf{u}_i$ being a d-dimensional representation vector of node i. Then $\mathbf{u}_i \cdot \mathbf{u}_j$ represents the expected similarity of node i and node j on their representation space. Nodes connected by higher weights are more likely to have similar vector representations. Since the adjacency matrix **G** is symmetric, we propose to approximate **G** with the product of **U** and $\mathbf{U}^\mathrm{T}$. By using the Frobenius norm as the cost function, we give the following optimization function:

$$\mathscr{L}_{\mathbf{G}} = ||\mathbf{G} - \mathbf{U}\mathbf{U}^\mathrm{T}||_\mathrm{F}. \tag{2}$$

To model the attributes, we investigate how to make the representation **U** also well preserve the node attribute proximity. We propose to approximate representation matrix **U** with the product of the attribute affinity matrix **A** and a weight matrix **W**. The intuition is that the nodes with similar representation can be characterized by the common attributes. Hence, we have the following optimization function:

$$\mathscr{L}_{\mathbf{A}} = ||\mathbf{U} - \mathbf{A}\mathbf{W}||_\mathrm{F}. \tag{3}$$

In addition, to prevent the values of some columns of **W** too large and impose sparseness constraints, we also invoke $l1$-regularization on **W**. Thus, our optimization function is:

$$\mathscr{L}_{\mathbf{A}} = ||\mathbf{U} - \mathbf{A}\mathbf{W}||_\mathrm{F} + \lambda \sum_{i=1}^{m} ||\mathbf{w}_i^\mathbf{T}||_1^2, \tag{4}$$

where $\mathbf{w}_i^T$ is the ith column vector of **W** and λ is a free parameter that controls the weight of the regularization term.

Due to the heterogeneity of networks, some marginal nodes that very close to the query node may not be well expressed because some hub nodes with large number of edges and attributes may account for most of the updates during the training. We expect that all the other nodes could be projected into the subspaces of the query node, so that the query node have more meaningful representation.

Input : $\mathcal{G}$, k, θ, query node q.
Output: A set $\mathcal{S}$ with k diversified nodes.

$\mathcal{S} \leftarrow \emptyset$
`/* Step 1: Learn node representations by NMF */`
Learn node representation matrix **U** through NMF
`/* Step 2: Perform diversification by MaxSum */`
while $|\mathcal{S}| < k$ **do**
    $v \leftarrow \max_{u \in \mathcal{V}} f_{\text{MaxSum}}(\mathcal{S})$
    $\mathcal{S} = \mathcal{S} \bigcup \{u\}$
    $\mathcal{V} = \mathcal{V} \backslash u$
return $\mathcal{S}$.

Algorithm 1. Overview of NMFDIV.

To address this issue, we use the similarity of nodes to the query node as the weight of balancing the updates between query and other nodes, which gives rise to the following objective function:

$$\mathscr{L}_q = \frac{1}{2}\sum_{i=1}^{n-1} ||\mathbf{u}_q - \mathbf{u}_i||_2^2 sim(q, i), \tag{5}$$

where $sim(q, i)$ is the similarity function measuring the similarity between pair nodes and can be acquired by many exiting similarity measures such as PPR (Jeh & Widom, 2002) and SimRank (Haveliwala, 2002). In this paper, we adopt Panther (Zhang et al., 2017) to evaluate $sim(q, i)$, since Panther has a lower time complexity than other algorithms. To make them complement each other, we jointly model the three types of information in the following optimization problem:

$$\min_{\mathbf{U},\mathbf{W}} \mathscr{L} = \min\{\mathscr{L}_{\mathbf{G}} + \alpha\mathscr{L}_{\mathbf{A}} + \beta\mathscr{L}_q\}, \tag{6}$$

where α and β are the free parameters trading off the three components.

4.3. Training NMF model via the multiplicative algorithm

Since the joint loss function in Eq. (6) is not convex, it is intractable to obtain the optimal solution. However, there are many algorithms from numerical optimization that can be applied to find local minima solutions. Gradient descent is perhaps the simplest technique to implement, but convergence can be slow (Lee & Seung, 2001). To address this issue, in this paper we assume that the representation matrix **U** and the weight matrix **W** are all nonnegative, so that we can use the *Multiplicative Algorithm* to achieve a local minima solution of Eq. (6) by using Majorization–Minimization framework (Hoyer, 2004). The nonnegative assumption is feasible since the relevance between node pairs also can be well measured in nonnegative subspaces. Our multiplicative algorithm iteratively updates **U** with **W** fixed and then **W** with **U** fixed, which guarantees not to increase the objective function after each iteration. In the following, we derive the updating rules of **U** and **W**.

Updating $\mathbf{u}_{ik}$. To derive the updating rule of $\mathbf{u}_{ik}$, we need to minimize the following objective function:

$$\mathscr{L}(\mathbf{U}) = ||\mathbf{G} - \mathbf{U}\mathbf{U}^{\mathrm{T}}||_{\mathrm{F}} + \alpha\, ||\mathbf{U} - \mathbf{A}\mathbf{W}||_{\mathrm{F}} + \frac{1}{2}\beta \sum_{i=1}^{n-1} ||\mathbf{u}_q - \mathbf{u}_i||_2^2 sim(q, i). \tag{7}$$

We let ψ_{ik} be the Lagrange multiplier for constraint $u_{ik} \ge 0$, and $\Psi = [\psi_{ik}]$. Introduce the Lagrange multipliers to our objective function resulting in the following equivalent objective function:

$$\begin{aligned}\mathscr{L}(\mathbf{U}) &= \mathrm{Tr}(\mathbf{G}\mathbf{G}^T - 2\mathbf{G}\mathbf{U}\mathbf{U}^T + \mathbf{U}\mathbf{U}^T\mathbf{U}\mathbf{U}^T)\\ &+ \alpha\,\mathrm{Tr}(\mathbf{U}\mathbf{U}^T - 2\mathbf{A}\mathbf{W}\mathbf{U}^T + \mathbf{A}\mathbf{W}\mathbf{W}^T\mathbf{A}^T)\\ &+ \beta \sum \mathrm{Tr}\left(\frac{1}{2}\mathbf{u}_i^t\mathbf{u}_i - \mathbf{u}_q^t\mathbf{u}_i\right) sim(q, i) + \mathrm{Tr}(\Psi\mathbf{U}^T).\end{aligned} \tag{8}$$

Then the derivative of $\mathscr{L}$ w.r.t. $\mathbf{u}_{ik}$ with **W** fixed is:

$$\frac{\partial\mathscr{L}}{\partial\mathbf{u}_{ik}} = (4\mathbf{U}\mathbf{U}^T\mathbf{U} - 4\mathbf{G}\mathbf{U} + \alpha 2\mathbf{U} - \alpha 2\mathbf{A}\mathbf{W})_{ik} + \beta\mathbf{u}_{ik} sim(q, i) + \boldsymbol{\psi}_{ik}. \tag{9}$$

Setting this derivative to zero, the Karush–Kuhn–Tucker (KKT) complementary slackness condition for the nonnegativity of **U** and **W** gives the following updating rule for $\mathbf{u}_{ik}$:

$$\mathbf{u}_{ik} \leftarrow \mathbf{u}_{ik}\,\mu_u, \tag{10}$$

where

$$\mu_u = \left(\frac{(4\mathbf{G}\mathbf{U} - \alpha 2\mathbf{U} + \alpha 2\mathbf{A}\mathbf{W})_{ik} + \beta\mathbf{u}_{ik} sim(q, i)}{(4\mathbf{U}\mathbf{U}^t\mathbf{U})_{ik}}\right)^{\frac{1}{4}}. \tag{11}$$

Updating $\mathbf{w}_{ik}$. To derive the updating rule of $\mathbf{w}_{ik}$, we need to minimize the following objective function:

$$\mathscr{L}(\mathbf{U}) = ||\mathbf{U} - \mathbf{A}\mathbf{W}||_{\mathrm{F}} + \lambda \sum_{i=1}^{m} ||\mathbf{w}_i^{\mathrm{T}}||_1^2, \tag{12}$$

which is equivalent to minimize the following objective function (Kim & Park, 2008):

$$\mathscr{L}(\mathbf{U}) = \left\|\begin{pmatrix}\mathbf{A}\\ \sqrt{\lambda}\ \mathbf{e}_1 \times \mathbf{m}\end{pmatrix}\mathbf{W} - \begin{pmatrix}\mathbf{U}\\ \mathbf{0}_1 \times \mathbf{d}\end{pmatrix}\right\|_{\mathrm{F}} = ||\mathbf{A}^*\mathbf{W} - \mathbf{U}^*||_{\mathrm{F}}, \tag{13}$$

where $\mathbf{e}_1 \times m$ is a row vector with all components equal to one and $\mathbf{0}_{1\times d}$ is a zero vector. Then following the same derivation of $\mathbf{u}_{ik}$, we can get the updating rule for $\mathbf{w}_{ik}$:

$$\mathbf{w}_{ik} \leftarrow \mathbf{w}_{ik}\,\mu_w, \tag{14}$$

Input : Adjacency matrix $\mathbf{G}$, attribute affinity matrix $\mathbf{A}$, query node q, parameters α, β and dimension d.
Output: Node representation matrix $\mathbf{U}$ and attribute weight matrix $\mathbf{W}$.

```
Initializing U and W
while not converge do
    Update U according to Eq. 10
    Update W according to Eq. 14
    Reinitializing zeros entries of U and W
return U and W.
```

Algorithm 2. Learn node representations by NMF.

Table 1
Statistics of real-world datasets.

Datasets	Nodes	Edges	Attributes	Aspects
Facebook	4039	88,234	1406	193
BlogCatalog	5196	171,743	8189	6
Flickr	7564	239,365	12,047	9
DBLP	73,242	373,797	122,043	172

where

$$\mu_w = \frac{(\mathbf{A}^{*T}\mathbf{U}^*)_{ik}}{(\mathbf{A}^{*T}\mathbf{A}^*\mathbf{W})_{ik}}. \tag{15}$$

With these updating rules, the multiplicative optimization algorithm for our NMF model is presented in Algorithm 2. Specifically, our algorithm first initializes **U** and **W** to be positive, then update them according to Eqs. (10) and (14) while reinitializing zeros entries of **U** and **W** to a small positive constant when $\mu_u \geq 1$ and $\mu_w \geq 1$. This simple reinitialize strategy prevents convergence to nonstationary point (Chi & Kolda, 2012). The following theorem shows that Algorithm 2 can converge to a stationary point.

Theorem 1. *The loss function 6 is nonincreasing and converges to a stationary point under the updating rules of Eqs. (10) and (14).*

Proof. See Appendix A. □

4.4. Complexity analysis

The majority cost of our multiplicative algorithm is the matrix multiplication part. The time complexity of multiplication for two matrices, e.g., a $n \times d$ matrix and a $d \times m$ matrix, is $O(ndm)$. Therefore, the time complexity for our algorithm is $O(rd(nm + n^2))$, where r is the iteration times, $d \ll n$ and $d \ll m$.

5. Experiments

In this section, we describe the datasets, specify our baselines, detail the metrics, and report the results.

5.1. Datasets

We consider four real-world network datasets with their brief statistical information listed in Table 1. The Facebook dataset is downloaded from SNAP[2] (Leskovec & Mcauley, 2012) and built from the profile and relation data of 10 users in Facebook, with their profiles being the attributes. There are 193 social circles in this network, which were manually identified by the 10 users via choosing which group (e.g. universities, colleagues and relatives) are their friends belonged to Leskovec and Mcauley (2012). We consider these social circles as the ground truth aspects of the 10 users. The BlogCatalog dataset (Huang et al., 2017a) is a social blogger community, in which users' interactions are viewed as the edges of the network. Users specify their keywords as a short description of their blogs, which are severed as node attributes. Users also register their blogs under predefined categories, which are viewed as the ground-truth aspects. The Flickr dataset (Huang et al., 2017a) is collected from Flickr.[3] Edges of this network are built form users friendship, and their group memberships are viewed as the ground-truth aspects. We also build a coauthor network extracted from the DBLP public bibliography data[4]. All the nodes of this network are authors who have published papers in top 172 computer science conferences (Tiers A and B ranked by CCF[5]), and all the edges represent the cooperations of authors in a same paper. We extract keyterms of these papers to be as attributes of nodes. All the 172 conferences are regarded as the ground-truth aspects. The four networks are undirected and their edge weights are set to 1.

To evaluate the scalability performance of our algorithm, we also generate a series of synthetic datasets with different size of nodes and attributes, according to the *ER* random network model (Erdös & Rényi, 1959).

5.2. Baselines

To evaluate the performance of our approach, we compare our approach with the state-of-the-art network search diversification approaches, which are introduced as follows.

- γ-RLM (Küçüktunç et al., 2015). This is a diversified citation recommendation method based on vertex selection and query

[2] http://snap.stanford.edu/data/
[3] http://socialcomputing.asu.edu/
[4] http://dblp.uni-trier.de/xml/
[5] http://www.ccf.org.cn/

refinement. This algorithm incrementally gets local maxima within the top-γk results until $|S| = k$, and removes the selected vertices from the subgraph for the next local maxima selection.

- Exp (Li & Yu, 2013). It is a diversified ranking algorithm for pure networks, which takes the *expansion ratio* as the diversity measure and optimizes a combined function of the relevance measure and the diversity measure. In this paper, we adopt its 2-step expansion as our comparison algorithm.
- RelEXP (Küçüktunç et al., 2013). This is a diversification algorithm for pure networks that combines both relevance and diversity into a single function. In this paper, we use its 2-step expanded relevance as our comparison algorithm.
- NMFDIV-NA. To validate the impact of attributes in diversified search task, we also consider the non-attribute version of NMFDIV, i.e., we remove the objective of Eq. (3).

5.3. Metrics

We evaluate our algorithm against several metrics, which are widely used in network diversification tasks (Küçüktunç et al., 2013; Li & Yu, 2013; Mei et al., 2010) and document diversification tasks (Zhai et al., 2003).

- *Density* (Mei et al., 2010): The *density* of the subgraph of the result set measures the number of edges in the subgraph divided by the maximum possible number of edges, and is calculated by:

$$Density(S_q^{(k)}) = \frac{2\sum_{u,v\in S_q^{(k)},\mathbf{g}_{uv}=1} 1}{k(k-1)}, \tag{16}$$

where $S_q^{(k)}$ is the top-k result nodes and $\mathbf{g}_{uv}$ is the edge weight of induced subgraph of node set $S_q^{(k)} \cap q$. Smaller density implies higher novelty of the result.

- *Group Coverage* (Li & Yu, 2013): It quantifies the number of aspects covered by the result nodes, calculated by:

$$Group\ Coverage(S_q^{(k)}) = \cup_{v\in S_q^{(k)}}(\mathscr{A}_q \cap \mathscr{A}_v), \tag{17}$$

where A_v is the number of aspects covered by v. This measure omits the actual intent of a query, and the group coverage of a same result set stays constant for any query.

- *S-recall* (Zhai et al., 2003): This metric quantifies the amount of unique aspects of the query q that are covered by the top k result nodes, according to:

$$S\text{-}recall(S_q^{(k)}) = \frac{\cup_{v\in S_q^{(k)}}(\mathscr{A}_q \cap \mathscr{A}_v)}{\mathscr{A}_q}. \tag{18}$$

S-recall is query-dependent and measures whether a covered aspect was actually useful or not.

Other metrics such as α-nDCG, MAP-IA and ERR-IA for evaluating the diversification performance in textual diversification researches are not applied for our evaluation, as the performance evaluated by them are essentially the same with that evaluated by the metrics used in this paper.

5.4. Results

5.4.1. Diversification performance

In this experiment, we randomly generate 100 queries, and compute the average of density, group coverage and S-recall for their query results. Figs. 2–4 report the performance comparison on our four real-world networks. Here the dimension d of NMFDIV is set to 128 for DBLP network and 64 for other three networks respectively.

Density. Fig. 2 compares the density performance with different result size k. In general, our proposed algorithms (NMFDIV, NMFDIV-NA) always achieve a relatively lower density in the four datasets. However, it is difficult to figure out which algorithm has a dominant performance among the four datasets. Recall that the lower density indicates the less similar to each other, which represents the result nodes are less redundant. This result implies that our diversification algorithm can more effectively achieve the novelty compared to other baselines.

Group coverage. Fig. 3 compares the group coverage performance by different result size k. We can find that RelEXP outperforms the other algorithms on group coverage metric in all the networks. Our NMFDIV achieves the second best performance in BlogCatalog and DBLP, but in other two networks it does not show any advantage in terms of this metric. RelEXP shows the best performance in Facebook, BlogCatalog and Flickr, while EXP shows the best performance in DBLP. This is not a surprising result, because with their optimization objectives of expansion relevance (Küçüktunç et al., 2013) and expansion ratio (Li & Yu, 2013), they tend to find the result set with a maximum domination of their neighbors, resulting in a good performance in group coverage. However, the high performance in group coverage does not indicate that the result satisfies the intents of the query.

S-recall. Fig. 4 shows the S-recall performance in our four networks by different result size k. From the four subfigures, we can observe that our NMFDIV significantly outperforms the baseline algorithms. RelEXP shows the relatively lower performance in S-recall among all the datasets, although it performs the best in group coverage metric, which suggests that results generated by RelEXP may be partly of no interest to the query. The high performance of NMFDIV is due to the fact that our approach have a clear objective

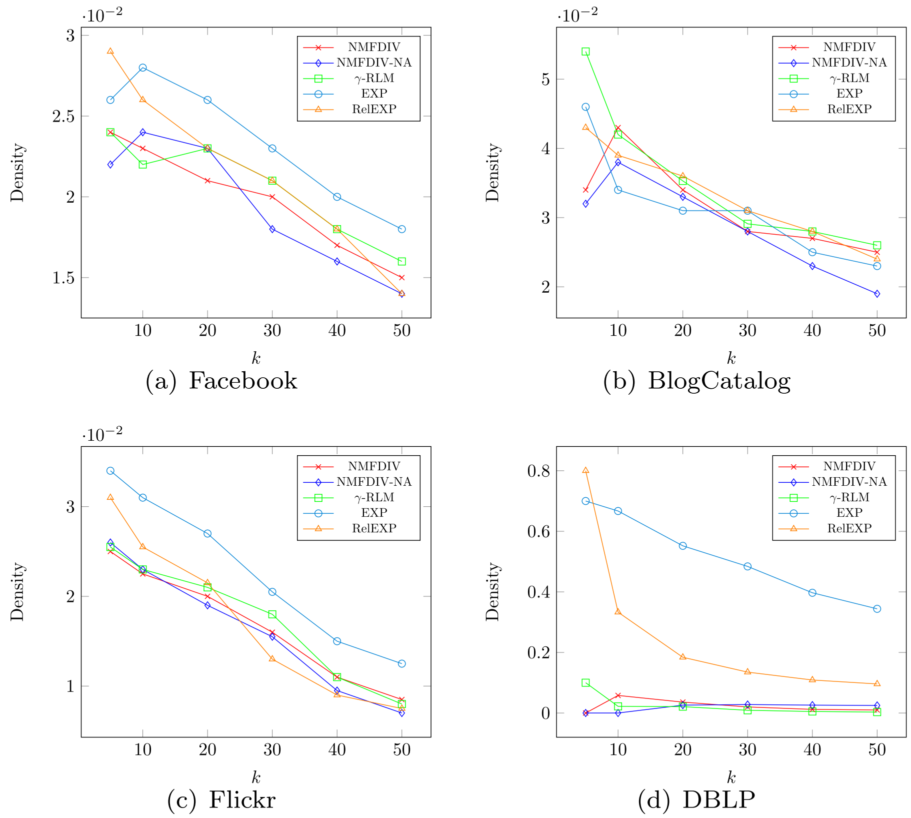


Fig. 2. Density performance by varying k.

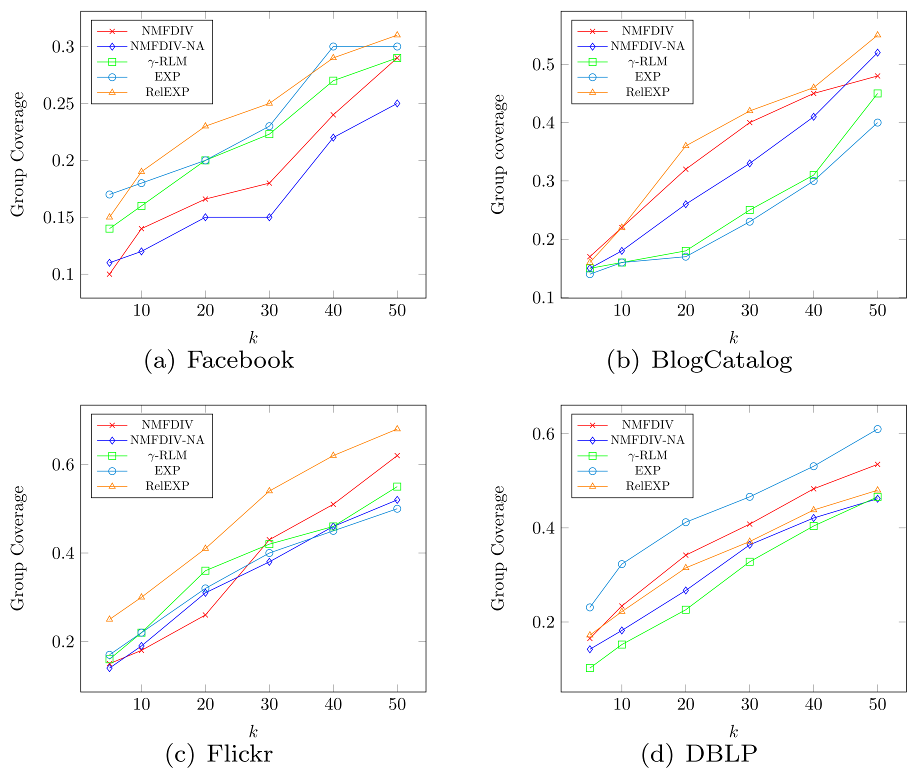


Fig. 3. Group coverage performance by varying k.

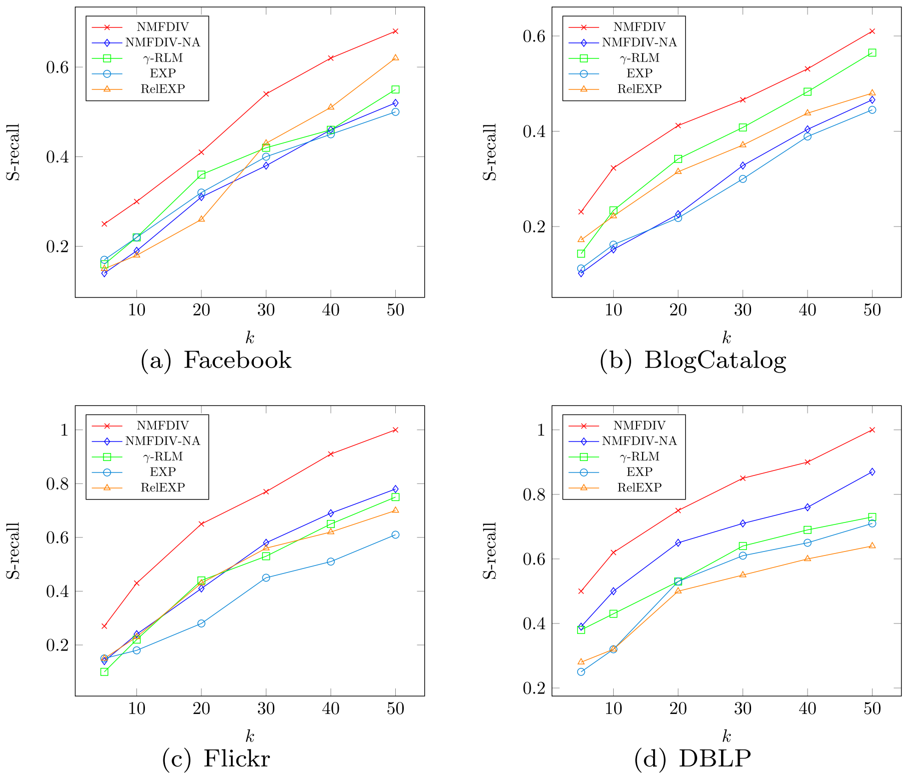


Fig. 4. S-recall performance by varying k.

to explicitly represent aspects of the query. On the other hand, we can see from Fig. 4 that NMFDIV always obtains a noticeable improvement on S-recall compared to NMFDIV-NA in all the networks, which means that integrating attributes do help improve the diversification performance.

5.4.2. Parameter analysis

In this subsection, we investigate the effects of the free parameters α, β and dimension d.

As discussed in our model, α balances the contributions of network structure and node attributes, while β indicates whether the representation of nodes is relevant to the query. We vary them from 0.1 to 1 simultaneously, and the result on DBLP is presented in Fig. 5(a). Similar results can be observed on other datasets. From the subfigure, we can observe that our method achieves relatively high performance when both α and β are in the middle, i.e., $\alpha = 0.6$ and $\beta = 0.6$. When $\beta = 0.1$, the best performance of S-recall is only 0.35 as α increased. But in the case of $\beta = 0.6$, the lowest performance of S-recall is 0.42 and 120% improvement can be observed as α increased. This demonstrates that objective function Eq. (5) have a large impact on NMFDIV. As a conclusion, NMFDIV could achieve relatively high performance by setting reasonable parameters of α and β. Significant and positive impacts are observed from α

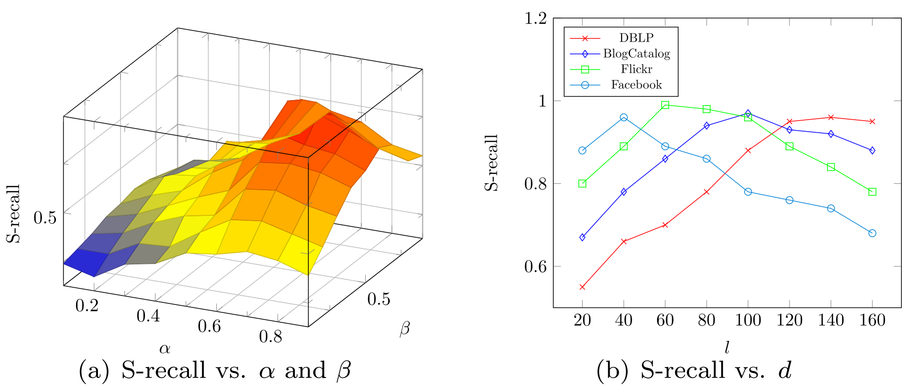
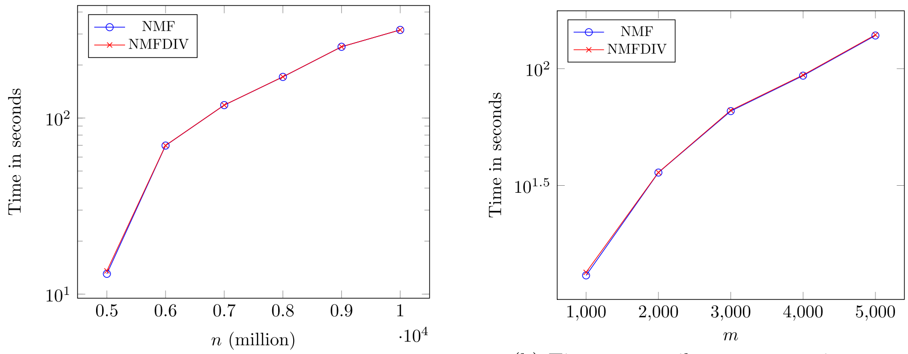


Fig. 5. S-recall performance on DBLP dataset w.r.t. free parameters α, β (see the subfigure 5(a)) and dimension d (see the subfigure 5(b)).



(a) Time v.s. node size on synthetic network.

(b) Time v.s. attribute category size on synthetic network.

Fig. 6. Scalability of NMFDIV.

and β, which verify that attributes and modeling query aspects play essential roles in the proposed method.

To study how the dimension d affects the diversification performance, we test the S-recall performance of NMFDIV on the four networks by varying dimension d from 20 to 180. Fig. 5(b) reports the result. We can see that too small d ($d \leq 20$) would result in inferior performance. The S-recall performance appears to increase as the growth of dimension and then peaks at different values of d with respect to different dataset. While our method reaches at the best S-recall performance in Facebook when $d = 40$, it reaches a peak when $d = 140$ in DBLP. This implies that more dimension should be taken for a larger dense network like DBLP to preserve features.

5.4.3. *Efficiency evaluation*

To evaluate the scalability performance of NMFDIV, we execute our algorithm on a series of synthetic datasets over different result size, with node size ranging from 5000 to 10,000, attribute category size ranging from 1000 to 5,000. The result is shown in Fig. 6. We can find that the performance curves of NMFDIV and NMF overlap with each other, which indicates that the MaxSum algorithm is quite efficient (linear w.r.t. n and k) compared to the network representation learning (i.e., NMF) part We can clearly see that the running time is almost linear to the node size and attribute category size if we keep the other factors unchanged. This result demonstrates the efficiency of the proposed method, and verifies the complexity analysis in Section 4.

5.4.4. *Case study*

Considering a query submitted to an academic social network platform, e.g., Aminer, the system would suggest candidate friends relevant to the result researcher q, and query users tend to focus on the first few recommendations. Intuitively, scientists often collaborate with their friends with multi-disciplinary fields of research, while query users may be only interested in one or a few research fields of them. The chances of successfully matching information needs of majority users will be higher if the system recommend a set of closest friends within different research fields. To understand the advantage of NMFDIV in such case, we conduct diversified search on a real coauthor network (Newman, 2006) with 379 scientists to find a diversified recommendations for researcher Mark Newman (the red circle). Fig. 7 shows the top-3 results of EXP, γ-RLM and NMFDIV, which are colored by green, purple and blue, respectively. We can observe that EXP returns Boccaletti, Barabasi and Jeong as recommendations which are all high degree nodes. This is because that EXP optimizes an objective of the neighbor expansion ratio. However, the recommendations of EXP are shown to be redundant, since Barabasi and Jeong are very similar and are affiliated in a same community. γ-RLM tends to find nodes that cover different communities, but some of such communities (e.g., the community Boccaletti is belonged to) seem to have no strong relevance to the query as the diversity metric of γ-RLM is query-independent. In contrast, our NMFDIV returns Solé, Stauffer and Kleinberg as the result. These scientists cover many research fields such as biological networks, complex networks, statistical physics and data mining, all of which are the main research fields of Newman. Therefore, we can draw the conclusion that NMFDIV improves the performance of recommendations by eliminating redundant results and covering multi-disciplinary fields of research w.r.t the query.

6. Conclusions

In this paper, we have proposed NMFDIV, a novel explicit search result diversification method for attributed networks, which consists of two components: learning node representation and diversification. Given a query node, our NMF approach projects an attributed network into a unified low dimension vector space w.r.t. the aspects of query node. These representation vectors are then used to estimate the relevance and the novelty of the result in order to select representative diverse nodes. In particular, our approach builds upon a general explicit diversification manner, which makes it seamlessly deployable by existing approaches in the literature.

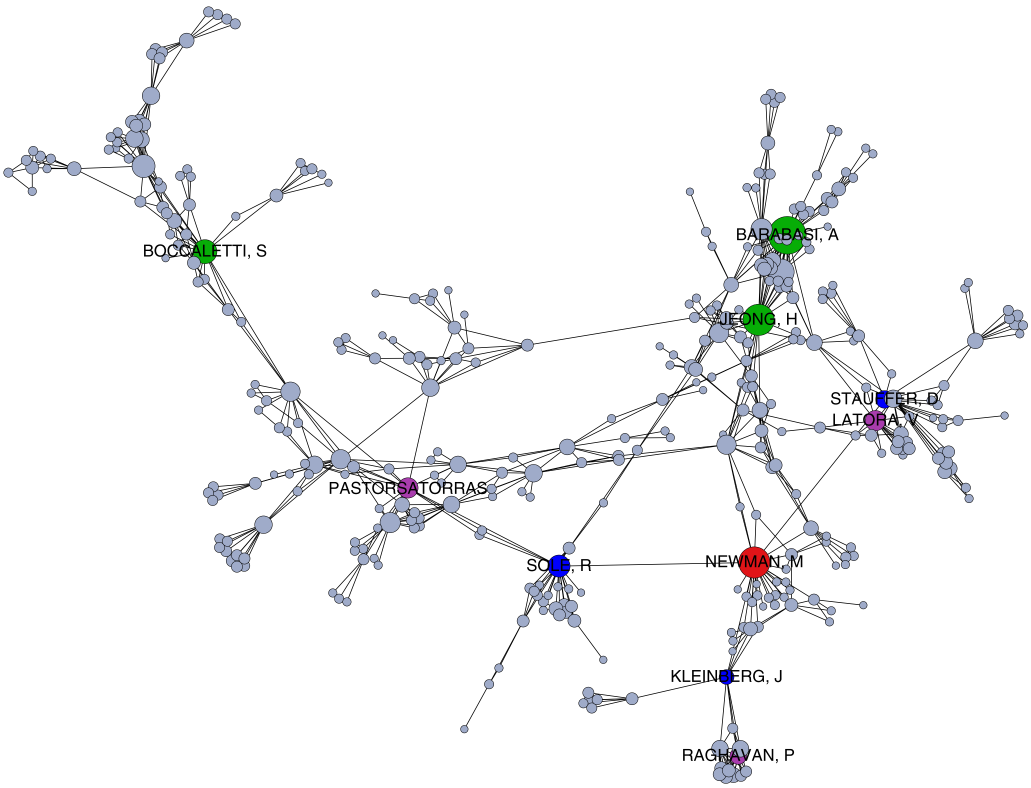


Fig. 7. A case study in a scientific co-author network. The top-3 results of researcher Mark Newman (in red) by EXP, γ-RLM and NMFDIV are colored by green, purple and blue, respectively. The degrees of nodes are indicated by the sizes of these circles. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Thorough experiments on real-world networks demonstrate effectiveness and efficiency of our algorithm. As to future work, we aim to incorporate the textual content and diffusion into our proposed method, which will be used to better capture the semanteme of each aspect.

Acknowledgment

This work is supported by National Key R & D Program of China Project #2017YFB0203201, Australian Research Council Discovery Project DP150104871 and National Natural Science Foundation of China (No. 61379157).

Appendix A. Proof of Theorem 1

To prove Theorem 1, we need to show that the optimization function of Eq. (6) is nonincreasing under the updating rules of Eqs. (10) and (14). We begin our proof with the definition of the auxiliary function.

Definition 1. (Lee & Seung, 2001) $G(\mathbf{U}, \mathbf{U}')$ is an auxiliary function for $F(\mathbf{U})$ if the conditions

$$G(\mathbf{U}, \mathbf{U}') \geq F(\mathbf{U}), G(\mathbf{U}, \mathbf{U}') = F(\mathbf{U}) \tag{A.1}$$

are satisfied.

Then the following lemmas hold.

Lemma 1. *(Lee & Seung, 2001) If G is an auxiliary function of F, then F is nonincreasing under the update*

$$\mathbf{U}^{(t+1)} = \arg\min_{\mathbf{U}} G(\mathbf{U}, \mathbf{U}^{(t)}). \tag{A.2}$$

Now, we show that Eq. (10) is exactly the update in Eq. (A.2) by defining the appropriate auxiliary functions for **U**.

Lemma 2. *The function*

$$\begin{aligned} G(\mathbf{U},\mathbf{U}') &= \mathrm{Tr}\big(\mathbf{G}\mathbf{G}^T - \mathbf{Z}^T\mathbf{G}'\mathbf{U}' - \mathbf{U}'^T\mathbf{G}'\mathbf{U}' + \mathbf{R}\mathbf{U}'^T\mathbf{U}'\mathbf{U}'^T\big)\\ &+ \mathrm{Tr}\big(\mathbf{U}'^T\mathbf{G}'\mathbf{Z} - 2\alpha\mathbf{W}^T\mathbf{A}^T\mathbf{Z} - 2\alpha\mathbf{W}^T\mathbf{A}^T\mathbf{U}' + \alpha\mathbf{A}\mathbf{W}\mathbf{W}^T\mathbf{A}^T\big)\\ &+ \beta\sum \mathrm{Tr}\left(\frac{1}{2}\mathbf{u}_i^t\mathbf{u}_i - \mathbf{u}_q^t\mathbf{u}_i\right)sim(q,i) \end{aligned} \tag{A.3}$$

*is an auxiliary function of function $\mathscr{L}(\mathbf{U})$ in Eq. (7), where $\mathbf{r}_{ij} = \frac{\mathbf{u}_{ij}^4}{\mathbf{u}_{ij}'^3}$, $\mathbf{z}_{ij} = \mathbf{u}_{ij}'\ln\frac{\mathbf{u}_{ij}}{\mathbf{u}_{ij}'}$, $\mathbf{G}' = 2\mathbf{G} - \alpha\mathbf{I}$, and **I** is an identity matrix.*

Proof. From Eq. (7), we have:

$$\begin{aligned} \mathscr{L}(\mathbf{U}) &= \mathrm{Tr}(\mathbf{G}\mathbf{G}^T - 2\mathbf{G}\mathbf{U}\mathbf{U}^T + \mathbf{U}\mathbf{U}^T\mathbf{U}\mathbf{U}^T)\\ &+ \alpha\,\mathrm{Tr}(\mathbf{U}\mathbf{U}^T - 2\mathbf{A}\mathbf{W}\mathbf{U}^T + \mathbf{A}\mathbf{W}\mathbf{W}^T\mathbf{A}^T)\\ &+ \beta\sum \mathrm{Tr}\left(\frac{1}{2}\mathbf{u}_i^t\mathbf{u}_i - \mathbf{u}_q^t\mathbf{u}_i\right)sim(q,i). \end{aligned} \tag{A.4}$$

It is straightforward to verify that $G(\mathbf{U},\mathbf{U}) = \mathscr{L}(\mathbf{U})$. To show that $G(\mathbf{U},\mathbf{U}') \geq \mathscr{L}(\mathbf{U})$, we need to prove that:

$$\begin{aligned} &-\mathrm{Tr}(\mathbf{Z}^T\mathbf{G}'\mathbf{U}') - \mathrm{Tr}\big(\mathbf{U}'^T\mathbf{G}'\mathbf{U}'\big) + \mathrm{Tr}\big(\mathbf{R}\mathbf{U}'^T\mathbf{U}'\mathbf{U}'^T\big)\\ &+ \mathrm{Tr}\big(\mathbf{U}'^T\mathbf{G}'\mathbf{Z}\big) - 2\alpha\,\mathrm{Tr}(\mathbf{W}^T\mathbf{A}^T\mathbf{Z}) - 2\alpha\,\mathrm{Tr}(\mathbf{W}^T\mathbf{A}^T\mathbf{U}')\\ &\geq -2\,\mathrm{Tr}(\mathbf{G}\mathbf{U}\mathbf{U}^T) + \mathrm{Tr}(\mathbf{U}\mathbf{U}^T\mathbf{U}\mathbf{U}^T) + \alpha\,\mathrm{Tr}(\mathbf{U}\mathbf{U}^T) - 2\alpha\,\mathrm{Tr}(\mathbf{A}\mathbf{W}\mathbf{U}^T). \end{aligned} \tag{A.5}$$

Since we have

$$\mathrm{Tr}\big(\mathbf{R}\mathbf{U}'^T\mathbf{U}'\mathbf{U}'^T\big) \geq \mathrm{Tr}(\mathbf{U}\mathbf{U}^T\mathbf{U}\mathbf{U}^T), \tag{A.6}$$

according to Lemma 5 and Lemma 7 of Wang, Li, Wang, Zhu, and Ding (2011),

$$\begin{aligned} &-2\,\mathrm{Tr}(\mathbf{G}\mathbf{U}\mathbf{U}^T) + \alpha\,\mathrm{Tr}(\mathbf{U}\mathbf{U}^T) = -\mathrm{Tr}(\mathbf{G}'\mathbf{U}\mathbf{U}^T)\\ &\leq -\mathrm{Tr}(\mathbf{Z}^T\mathbf{G}'\mathbf{U}') - \mathrm{Tr}\big(\mathbf{U}'^T\mathbf{G}'\mathbf{U}'\big) + \mathrm{Tr}\big(\mathbf{U}'^T\mathbf{G}'\mathbf{Z}\big), \end{aligned} \tag{A.7}$$

according to Lemma 4 of Wang et al. (2011), and

$$-\mathrm{Tr}(\mathbf{A}\mathbf{W}\mathbf{U}^T) \leq -\mathrm{Tr}(\mathbf{W}^T\mathbf{A}^T\mathbf{Z}) - \mathrm{Tr}(\mathbf{W}^T\mathbf{A}^T\mathbf{U}'), \tag{A.8}$$

according to Lemma 2 of Wang et al. (2011), Eq. (A.5) holds. Hence, $G(\mathbf{U},\mathbf{U}') \geq \mathscr{L}(\mathbf{U})$.
□

Proof of Theorem 1.

Proof. Lemma 2 provides a specific form $G(\mathbf{U},\mathbf{U}')$ of the auxiliary function for $\mathscr{L}(\mathbf{U})$ in Eq. (7). Then we can find the solution for $\arg\min_{\mathbf{U}} G(\mathbf{U},\mathbf{U}^{(t)})$ by the following KKT condition:

$$\frac{\partial G(\mathbf{U},\mathbf{U}')}{\partial \mathbf{u}_{ik}} = 4\big(\mathbf{U}'\mathbf{U}'^T\mathbf{U}'\big)_{ik}\frac{\mathbf{u}_{ik}^3}{\mathbf{u}_{ik}'^3} - (2\mathbf{G}'\mathbf{U}' + 2\alpha\mathbf{A}\mathbf{W})_{ik}\frac{\mathbf{u}_{ik}}{\mathbf{u}_{ik}'}, \tag{A.9}$$

which gives rise to the updating rule in Eq. (10). According to Lemma 2, $\mathscr{L}(\mathbf{U})$ is nonincreasing.

$\mathscr{L}(\mathbf{W})$ can similarly be shown to be nonincreasing under the update rules for **W**. Hence, Theorem 1 holds. □

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.ipm.2018.05.005.

References

Agrawal, R., Gollapudi, S., Halverson, A., & Ieong, S. (2009). *Diversifying search results. WSDM*. ACM5–14.
Borodin, A., Lee, H. C., & Ye, Y. (2012). *Max-sum diversification, monotone submodular functions and dynamic updates. PODS*. ACM155–166.
Carbonell, J., & Goldstein, J. (1998). *The use of mmr, diversity-based reranking for reordering documents and producing summaries. SIGIR*. ACM335–336.
Chakraborty, Y. P. N., & Sycara, K. (2015). *Nonnegative matrix tri-factorization with graph regularization for community detection in social networks. AAAI*. AAAI Press2083–2089.
Chen, H., & Karger, D. R. (2006). *Less is more: Probabilistic models for retrieving fewer relevant documents. SIGIR*. ACM429–436.
Cheng, H., Zhou, Y., & Yu, J. X. (2011). Clustering large attributed graphs: A balance between structural and attribute similarities. *TKDD, 5*(2), 12.

Chi, E., & Kolda, T. (2012). On tensors, sparsity, and nonnegative factorizations. *SIAM Journal on Matrix Analysis and Applications, 33*(4), 12721299.
Dang, V., & Croft, W. B. (2012). *Diversity by proportionality: An election-based approach to search result diversification. SIGIR*. ACM65–74.
Deng, C., Xiaofei, H., Jiawei, H., & Huang, T. S. (2011). Graph regularized nonnegative matrix factorization for data representation. *TPAMI, 33*(8), 1548–1560 2011-08
Erdös, P., & Rényi, A. (1959). On random graphs i. *Publicationes Mathematicae Debrecen, 6*, 290–297.
Gollapudi, S., & Sharma, A. (2009). *An axiomatic approach for result diversification. WWW*. ACM381–390.
Haveliwala, T. H. (2002). *Topic-sensitive pagerank. WWW*. ACM517–526.
He, J., Hollink, V., & de Vries, A. (2012). *Combining implicit and explicit topic representations for result diversification. SIGIR*. ACM851–860.
Hoyer, P. O. (2004). Non-negative matrix factorization with sparseness constraints. *JMLR, 5*, 1457–1469.
Huang, X., Li, J., & Hu, X. (Li, Hu, 2017a). *Accelerated attributed network embedding. ICDM*. SIAM633–641.
Huang, X., Li, J., & Hu, X. (Li, Hu, 2017b). *Label informed attributed network embedding. WSDM*. ACM731–739.
Jeh, G., & Widom, J. (2002). *Simrank: A measure of structural-context similarity. SIGKDD*. ACM538–543.
Kim, J., & Park, H. (2008). *Sparse nonnegative matrix factorization for clustering.* Georgia Institute of Technology1–15.
Küçüktunç, O., Saule, E., Kaya, K., & Çatalyürek, Ü. V. (2013). *Diversified recommendation on graphs: Pitfalls, measures, and algorithms. WWW*. ACM715–726.
Küçüktunç, O., Saule, E., Kaya, K., & Çatalyürek, Ü. V. (2015). Diversifying citation recommendations. *TIST, 5*(4), 55.
Lee, D. D., & Seung, H. S. (2001). *Algorithms for non-negative matrix factorization. NIPS*. MIT Press556–562.
Lee, D. H., & Brusilovsky, P. (2017). Improving personalized recommendations using community membership information. *Information Processing & Management, 53*(5), 12011214.
Leskovec, J., & Mcauley, J. J. (2012). *Learning to discover social circles in ego networks. NIPS*. Curran Associates, Inc539–547.
Li, R.-H., & Yu, J. X. (2013). Scalable diversified ranking on large graphs. *TKDE, 25*(9), 2133–2146.
Liang, S., Cai, F., Ren, Z., & de Rijke, M. (2016). Efficient structured learning for personalized diversification. *IEEE Transactions on Knowledge and Data Engineering, 28*(11), 2958–2973.
Liang, S., Ren, Z., & De Rijke, M. (Ren, De Rijke, 2014a). *Fusion helps diversification. SIGIR*. ACM303–312.
Liang, S., Ren, Z., & De Rijke, M. (Ren, De Rijke, 2014b). *Personalized search result diversification via structured learning. SIGKDD*. ACM751–760.
Liang, S., & de Rijke, M. (2015). Burst-aware data fusion for microblog search. *Information Processing & Management, 51*(2), 89–113.
Liang, S., & de Rijke, M. (2016). Formal language models for finding groups of experts. *Information Processing & Management, 52*(4), 529–549.
Liang, S., Yilmaz, E., Shen, H., de Rijke, M., & Croft, W. B. (2017). Search result diversification in short text streams. *ACM Transactionson Information Systems, 36*(1), 8:1–8:35.
Mei, Q., Guo, J., & Radev, D. (2010). *Divrank: The interplay of prestige and diversity in information networks. SIGKDD*. ACM1009–1018.
Newman, M. E. J. (2006). Finding community structure in networks using the eigenvectors of matrices. *Physical Review E, 74*(3).
Psorakis, I., Roberts, S., Ebden, M., & Sheldon, B. (2011). Overlapping community detection using bayesian non-negative matrix factorization. *Physical Review E, 83*, 066114.
Santos, R. L., Macdonald, C., & Ounis, I. (2010). *Exploiting query reformulations for web search result diversification. WWW*. ACM881–890.
Santos, R. L., Macdonald, C., & Ounis, I. (2011). *Intent-aware search result diversification. SIGIR*. ACM595–604.
Seung, H. S., & Lee, D. D. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature, 401*, 789–792.
Tong, H., He, J., Wen, Z., Konuru, R., & Lin, C.-Y. (2011). *Diversified ranking on large graphs: An optimization viewpoint. SIGKDD*. ACM1028–1036.
Wang, F., Li, T., Wang, X., Zhu, S., & Ding, C. (2011). Community discovery using nonnegative matrix factorization. *Data Mining and Knowledge Discovery, 22*(3), 493–521.
Wang, J., & Zhu, J. (2009). *Portfolio theory of information retrieval. SIGIR*. ACM115–122.
Wang, Y.-X., & Zhang, Y.-J. (2013). Nonnegative matrix factorization: A comprehensive review. *TKDE, 25*(6), 1336–1353 2013-06
Wu, J., Zhang, G., & Ren, Y. (2017). A balanced modularity maximization link prediction model in social networks. *Information Processing & Management, 53*(1), 295307.
Xu, W., Liu, X., & Gong, Y. (2003). *Document clustering based on non-negative matrix factorization. SIGIR*. ACM267–273 01359
Zhai, C. X., Cohen, W. W., & Lafferty, J. (2003). *Beyond independent relevance: Methods and evaluation metrics for subtopic retrieval. SIGIR*. ACM10–17.
Zhang, J., Tang, J., Ma, C., Tong, H., Jing, Y., Li, J., et al. (2017). Fast and flexible top-k similarity search on large networks. *TOIS, 36*(2).
Zhang, J., Yu, P. S., & Zhou, Z.-H. (2014). *Meta-path based multi-network collective link prediction. SIGKDD*. ACM1286–1295.