Asymmetric Semi-Nonnegative Matrix Factorization for Directed Graph Clustering

Reyhaneh Abdollahi
Department of Computer Engineering
Islamic Azad University, Ahvaz Branch
Ahvaz, Iran
reyhanehabdollahi@outlook.com

Seyed Amjad Seyedi
Department of Computer Engineering
University of Kurdistan
Sanandaj, Iran
amjadseyedi@eng.uok.ac.ir

Mohamad Reza Noorimehr
Department of Computer Engineering
Islamic Azad University, Ahvaz Branch
Ahvaz, Iran
r.noorimehr@iauahvaz.ac.ir

***Abstract*—Graph clustering is a fundamental task in the network analysis, which is essential for many modern applications. In recent years, Nonnegative Matrix Factorization (NMF) has been effectively used to discover cluster structures due to its powerful interpretability property. In this paper, we introduce a clustering algorithm based on Semi-Nonnegative Matrix Factorization that is one of the well-known extensions of NMF. This factorization allows algorithms to capture more accurate (positive and negative) relationships among clusters and, thereby, to derive a latent factor that is even proper for clustering and also has much more responsibility in the regularization. Moreover, to improve the clustering, we define an asymmetric graph regularization to penalize the asymmetric similarity of nodes denoted by cluster memberships. Experimental results on four real-world datasets validate the effectiveness of the proposed method.**

***Keywords*—Directed networks; Graph clustering; Semi-Nonnnegative Matrix Factorization; Asymmetric similarity; Asymmetric regularization**

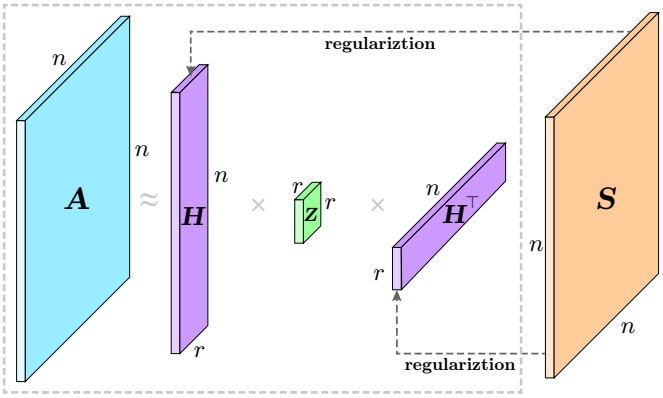

Fig. 1. Diagram of a regularized asymmetric nonneagtive matrix factorization method that decomposes a asymmetric adjacency matrix into two matrices consisting of cluster membership and cluster relationship.

I. Introduction

Complex networks have been considered as an important subject of discussion for scientists and researchers and attracts an increasing interest in recent years. These networks contain an expansive variety of systems in nature and society; systems composed by a great number of highly interconnected entities. These networks comprise social networks, citation networks, technological networks, large circuits, power networks, transportation networks, information networks, etc [1].

Complex networks can have community structure if the objects can be assigned into sets of nodes as groups where each group has internal communications. studies that apply clustering for network analysis are significantly common. clustering is the task of partitioning objects into logical categories that called clusters. Clustering problems can be distinct into two ways of evolve. The first and the popular one is grouping the entities of a single graph based on their similarity and the other one aims to cluster a set of graphs by assuming each of them as a sole object. In this approach, a cluster as a group of nodes that are more similar to each other compared to the rest of the nodes in the network is defined. Moreover, Inter-cluster correlation is high and clusters are connected with the minimum number of links. A well-organized clustering method gives us a better understanding of a given network [2].

Most of the research so far has studied the undirected networks. This is partially because directed networks bear more complicated connectivity. according to that, clustering in directed networks has become a more challenging task in contrast to undirected ones. Due to this, this paper focuses on finding clusters in graphs with directed edges between nodes [3].

In recent years several methods have been introduced for matrix factorization in the pattern analysis area [4]. Among these methods, nonnegative matrix factorization has been more popular and it is widely applied for the representation of nonnegative data such as pattern recognition [5], computer vision [6], information retrieval [7], and speech recognition [8]. When a set of n-dimensional data vectors is given as a matrix with just nonnegative elements, NMF is approximately factorized the primary matrix and detect the lower rank estimation that can be issued [9].

A variety of NMF-based clustering methods have been proposed recently [10]. The authors of [11] analyzed and presented three types of NMF methods. asymmetric nonnegative matrix factorization (ANMF) is one of the techniques that is developed for clustering in directed graphs using the adjacency matrix. Considering the simplicity of the adjacency matrix, some of the substantial data such as similarity and multiple links will be

978-1-7281-8566-8/20/$31.00 ©2020 IEEE

ignored. Moreover, in cases with large datasets, the algorithm will undergo high computational complexity because of the random initialization.

In [12], Tosyali et al. proposed regularized asymmetric nonnegative matrix factorization (RANMF) to perform the clustering task according to the mutual similarity of the nodes. This technique applied specified statistics of the nodes as an additional regularization factor on ANMF and offer a new method for matrix factorization.

In this paper, we propose a method called Asymmetric semi-NMF (in short AsNMF) that extends ANMF in such a way that exploits the positive and negative cluster correlations. We further developed our model to incorporate asymmetric similarity information and proposed the regularized RAsNMF method. Experiments are carried out on real-world datasets, and the results show the effectiveness of the proposed method.

The rest of this paper is organized below: Section II summarizes the Asymmetric Nonnegative matrix factorization on graph clustering. The proposed framework is described in detail in Section III. Experimental results on both synthetic and real-world datasets are presented in Section IV. Finally, the conclusion is provided in Section V.

II. Asymmetric NMF

It is clear that in directed networks, the edges included are all directed, which leads to the fact that adjacency matrix $\boldsymbol{A}$ is asymmetric and $\boldsymbol{A}_{ij} \neq \boldsymbol{A}_{ji}$. Adjacency matrix $\boldsymbol{A}$ defined as $\boldsymbol{A} \in \mathbb{R}_{+}^{n \times n}$ which n assigned as the number of nodes and $\boldsymbol{A}_{ij}$ is 1 if there exists a directed edge that connects node i to node j, and 0 otherwise. The ANMF algorithm performs the factorization by applying the adjacency matrix $\boldsymbol{A}$ and partitions the observations into c clusters [11].

Asymmetric nonnegative matrix factorization can be specified by the optimization problem below:

$$\min_{\boldsymbol{H} \geq 0, \boldsymbol{Z} \geq 0} \left\| \boldsymbol{A}_{+} - \boldsymbol{H}_{+} \boldsymbol{Z}_{+} \boldsymbol{H}_{+}^{\top} \right\|_{F}^{2}, \tag{1}$$

where $\left\| \cdot \right\|_{F}$ is the Frobenius norm and $\boldsymbol{H}$, $\boldsymbol{Z}$ defined as $\boldsymbol{H} \in \mathbb{R}_{+}^{n \times c}$, $\boldsymbol{Z} \in \mathbb{R}_{+}^{c \times c}$.

In the factorization shown above, $\boldsymbol{H}$ represents the information about cluster membership and the ith row in $\boldsymbol{H}$ indicates the Estimation of the ith observation insight of belonging to kth cluster for $k = 1, 2, \ldots, c$. Since this strategy works on directed networks and $\boldsymbol{A}$ is asymmetric, $\boldsymbol{Z}$ is also asymmetric. $\boldsymbol{Z}$ contains inter-cluster information, and the elements included demonstrate the directional connectivity within each cluster.

In the ANMF method, the initial values for $\boldsymbol{H}$ and $\boldsymbol{Z}$ will be randomly conducted, and the multiplicative updating formulas of matrices for this optimization problem are as below:

$$\boldsymbol{H} \leftarrow \boldsymbol{H} \odot \left[\frac{\boldsymbol{A}^{\top} \boldsymbol{H} \boldsymbol{Z} + \boldsymbol{A} \boldsymbol{H} \boldsymbol{Z}^{\top}}{\boldsymbol{H} \boldsymbol{Z}^{\top} \boldsymbol{H}^{\top} \boldsymbol{H} \boldsymbol{Z} + \boldsymbol{H} \boldsymbol{Z} \boldsymbol{H}^{\top} \boldsymbol{H} \boldsymbol{Z}^{\top}} \right]^{\frac{1}{4}}, \tag{2}$$

$$\boldsymbol{Z} \leftarrow \boldsymbol{Z} \odot \frac{\boldsymbol{H}^{\top} \boldsymbol{A} \boldsymbol{H}}{\boldsymbol{H}^{\top} \boldsymbol{H} \boldsymbol{Z} \boldsymbol{H}^{\top} \boldsymbol{H}}. \tag{3}$$

Despite the fact that ANMF algorithm generated a successful optimization for clustering in directed networks, in more detail just using adjacency matrix $\boldsymbol{A}$ and not giving consideration to the connectivity of the nodes belong to the same cluster in the approximated clustering in each iteration may result the existence of non-connected nodes within some clusters. Moreover, the random initialization for $\boldsymbol{H}$ and $\boldsymbol{Z}$ can issue misleading clustering and cause the converging of the updating algorithm on the local minimum. Also, initial estimates that generated randomly may require more iterations and consequently more computational cost.

III. Asymmetric Semi-NMF

In this section, we start by presenting the Asymmetric Semi-NMF (AsNMF) algorithm for the aim of clustering in directed graphs like the ANMF algorithm and in order to improve the clustering accuracy, we introduce a graph regularization factor using an efficient asymmetric similarity.

A. Basic model

Semi-NMF [13] relaxes the non-negativity constraint of ANMF and allows the cluster matrix $\boldsymbol{Z}$ to have positive and negative entries while it restricts only the membership matrix $\boldsymbol{H}$ so that it comprises of strictly non-negative components. It thus approximates the following cost function.

$$\min_{\boldsymbol{H} \geq 0} \mathcal{C} = \left\| \boldsymbol{A}_{\pm} - \boldsymbol{H}_{+} \boldsymbol{Z}_{\pm} \boldsymbol{H}_{+}^{\top} \right\|_{F}^{2}. \tag{4}$$

The ANMF model (1) exploits positive intra-cluster correlations, and it does not allow negative intra-cluster correlations. To cover this shortcoming, the AsNMF model (4) is used, which has the ability to exploit the positive and negative correlations among clusters. In this model, higher positive value z_{ij}^{+} representing stronger connectivity (or positive correlation) between cluster i and cluster j and the lower negative value z_{ij}^{-} representing stronger anti-connectivity (or negative correlation) between cluster i and cluster j, and this non-negativity improves the performance of graph clustering.

B. Asymmetric regularization

In this section, we use a regularization term in order to improve our clustering method in directed graph. This algorithm tries to capture mutual similarities among the nodes in the network by adding a regularization term to factorization algorithm.

We can suppose that a group of nodes belong to the same cluster should be close to each other and also far from the nodes in different clusters, so we can make an agreement on this, the closer the nodes are, the higher similarity we can get and it is more likely for nodes to be in the same cluster [14]. In order to reduce the possibility of existence of similar nodes in different clusters, in the regularization term that we have enhanced on factorization algorithm, the distance between nodes and also a similarity factor which we clarify in the following, are the primary statistics. This regulation is illustrated as follow:

$$\min_{\boldsymbol{H}} \sum_{i=1}^{n} \sum_{j=1}^{n} \left\| \boldsymbol{h}_i - \boldsymbol{h}_j \right\|_{2}^{2} \boldsymbol{S}_{ij}, \tag{5}$$

where $\boldsymbol{S}_{ij}$ shows the similarity of nodes (node i and node j) and $\left\|\boldsymbol{h}_i - \boldsymbol{h}_j\right\|_2$ indicates the distance between two nodes (node i and node j).

The space between nodes in a graph shows the closeness between them. Nodes belong to the same cluster should be closer to each other in comparison to ones in different clusters, in other words, they have a smaller distance, so we apply Euclidean distance to reach the value of closeness between two nodes [15]. So far, several techniques have been analyzed for calculation of similarity matrix $\boldsymbol{S}$. In this paper, we introduce two ways to obtain that. The first one and the easiest way to show the similarity between nodes is the adjacency matrix. Clearly, $\boldsymbol{A}_{ij}$ is the directed link that connects node i to node j.

The second method for evaluating the similarity between two nodes is based on the total number of mutual neighbors. Cosine similarity measures the matrix $\boldsymbol{S}$ through the following order:

$$cos(\boldsymbol{u}, \boldsymbol{v}) := \boldsymbol{S}_{1/2}(\boldsymbol{u}, \boldsymbol{v}) = \frac{\boldsymbol{u}^\top \boldsymbol{v}}{\left\|\boldsymbol{u}\right\| . \left\|\boldsymbol{v}\right\|}, \tag{6}$$

where u, v are the vectors of the adjacency matrix A, and $cos(\boldsymbol{u}, \boldsymbol{v}) = cos(\boldsymbol{v}, \boldsymbol{u})$.

Although a graph is successfully clustered by the Regularized ANMF (with symmetric similarity), the intrinsic characteristic of directed graphs is overlooked. Inspired by the assumption that directed graph may better capture the nature of network [16], an asymmetric weight measure, namely asymmetric cosine similarity [17], is introduced in order to define an asymmetric regularization and it can be shown as

$$acos(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{S}_{\alpha}(\boldsymbol{u}, \boldsymbol{v}) = \frac{\boldsymbol{u}^\top \boldsymbol{v}}{\left\|\boldsymbol{u}\right\|^{2\alpha} . \left\|\boldsymbol{v}\right\|^{2(1-\alpha)}}, \tag{7}$$

where $0 \leq \alpha \leq 1$ is introduced to control the intensity of asymmetry and by getting far from 0.5, the degree of asymmetry of $\boldsymbol{S}_{\alpha}$ increases.

In different situations of algorithm evaluation, we may use each similarity method for measuring the mutual similarity between nodes, so both of them are effective. According to the description given on the practical parameters that are applied in regularization, our algorithm seeks to perform the optimization based on the formulation below:

$$\begin{aligned} \mathcal{R} &= \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \left\|\boldsymbol{h}_i - \boldsymbol{h}_j\right\|_2^2 \boldsymbol{S}_{ij} \\ &= \sum_{i=1}^{n} \boldsymbol{h}_i^\top \boldsymbol{h}_i \boldsymbol{D}_{ii} - \sum_{i=1}^{n} \sum_{j=1}^{n} \boldsymbol{h}_i^\top \boldsymbol{h}_j \boldsymbol{S}_{ij} \\ &= \mathrm{Tr}(\boldsymbol{H}^\top \boldsymbol{D} \boldsymbol{H}) - \mathrm{Tr}(\boldsymbol{H}^\top \boldsymbol{S} \boldsymbol{H}), \end{aligned} \tag{8}$$

where $\mathrm{Tr}(\cdot)$ is the trace of a matrix and $\boldsymbol{D}$ is a diagonal matrix, which is the summation of ith row of $\boldsymbol{S}$ and $\boldsymbol{D}_{ii} = \sum_{j=1}^{n} \boldsymbol{S}_{ij}$.

By adding the regularization term (8) to the basic model (4) of the problem ($\mathcal{L} = \mathcal{C} + \lambda \mathcal{R}$) we can rewrite the optimization problem as

$$\begin{aligned} \min_{\boldsymbol{H} \geq 0} \mathcal{L} = &\left\|\boldsymbol{A}_{\pm} - \boldsymbol{H}_{+} \boldsymbol{Z}_{\pm} \boldsymbol{H}_{+}^\top\right\|_F^2 \\ &+ \lambda \mathrm{Tr}(\boldsymbol{H}^\top \boldsymbol{D} \boldsymbol{H}) - \lambda \mathrm{Tr}(\boldsymbol{H}^\top \boldsymbol{S} \boldsymbol{H}). \end{aligned} \tag{9}$$

C. Optimization

The optimization problem in (9) is not convex over the variables $\boldsymbol{Z}$ and $\boldsymbol{H}$ simultaneously. Thus, in this subsection, we employ an iterative optimization algorithm using the multiplicative update rules for local optimal solutions. The variables $\boldsymbol{Z}$ and $\boldsymbol{H}$ are alternately updated by fixing other variables. By rewriting the cost function (9) to the trace form, the following derivable equation is obtained.

$$\begin{aligned} \mathcal{L} &= \left\|\boldsymbol{A} - \boldsymbol{H} \boldsymbol{Z} \boldsymbol{H}^\top\right\|_F^2 + \lambda \mathrm{Tr}(\boldsymbol{H}^\top \boldsymbol{D} \boldsymbol{H}) - \lambda \mathrm{Tr}(\boldsymbol{H}^\top \boldsymbol{S} \boldsymbol{H}) \\ &= \mathrm{Tr}(\boldsymbol{A}^\top \boldsymbol{A} - 2 \boldsymbol{A}^\top \boldsymbol{H} \boldsymbol{Z} \boldsymbol{H}^\top + \boldsymbol{H} \boldsymbol{Z} \boldsymbol{H}^\top \boldsymbol{H} \boldsymbol{Z}^\top \boldsymbol{H}^\top) \\ &\quad + \lambda \mathrm{Tr}(\boldsymbol{H}^\top \boldsymbol{D} \boldsymbol{H}) - \lambda \mathrm{Tr}(\boldsymbol{H}^\top \boldsymbol{S} \boldsymbol{H}) - \mathrm{Tr}(\boldsymbol{\beta} \boldsymbol{H}^\top). \end{aligned} \tag{10}$$

Fixing $\boldsymbol{H}$, the solution for $\boldsymbol{Z}$ is obtained by computing

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{Z}} = -2 \boldsymbol{H}^\top \boldsymbol{A} \boldsymbol{H} + 2 \boldsymbol{H}^\top \boldsymbol{H} \boldsymbol{Z} \boldsymbol{H}^\top \boldsymbol{H} = 0. \tag{11}$$

This gives the solution

$$\boldsymbol{Z} \leftarrow (\boldsymbol{H}^\top \boldsymbol{H})^{-1} \boldsymbol{H}^\top \boldsymbol{A} \boldsymbol{H} (\boldsymbol{H}^\top \boldsymbol{H})^{-1} \tag{12}$$

or

$$\boldsymbol{Z} \leftarrow \boldsymbol{H}^{\dagger} \boldsymbol{A} (\boldsymbol{H}^\top)^{\dagger}, \tag{13}$$

where $\boldsymbol{H}^{\dagger}$ is the Moore–Penrose pseudo-inverse of $\boldsymbol{H}$.

Then the derivative of $\mathcal{L}$ with respect to $\boldsymbol{H}$ is:

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \boldsymbol{H}} = &-\boldsymbol{A}^\top \boldsymbol{H} \boldsymbol{Z} - \boldsymbol{A} \boldsymbol{H} \boldsymbol{Z}^\top \\ &+ 2 \boldsymbol{H} \boldsymbol{Z}^\top \boldsymbol{H}^\top \boldsymbol{H} \boldsymbol{Z} + 2 \boldsymbol{H} \boldsymbol{Z} \boldsymbol{H}^\top \boldsymbol{H} \boldsymbol{Z}^\top \\ &+ 2 \lambda \boldsymbol{D} \boldsymbol{H} - \lambda \boldsymbol{S} \boldsymbol{H} - \lambda \boldsymbol{S}^\top \boldsymbol{H} - \boldsymbol{\beta} = \boldsymbol{0}, \end{aligned} \tag{14}$$

where the Lagrangian multipliers $\boldsymbol{\beta}$ enforce nonnegative constraints and,

$$\begin{aligned} \boldsymbol{\beta} = &-\boldsymbol{A}^\top \boldsymbol{H} \boldsymbol{Z} - \boldsymbol{A} \boldsymbol{H} \boldsymbol{Z}^\top \\ &+ 2 \boldsymbol{H} \boldsymbol{Z}^\top \boldsymbol{H}^\top \boldsymbol{H} \boldsymbol{Z} + 2 \boldsymbol{H} \boldsymbol{Z} \boldsymbol{H}^\top \boldsymbol{H} \boldsymbol{Z}^\top \\ &+ 2 \lambda \boldsymbol{D} \boldsymbol{H} - \lambda \boldsymbol{S} \boldsymbol{H} - \lambda \boldsymbol{S}^\top \boldsymbol{H}. \end{aligned} \tag{15}$$

From the complementary slackness condition of the Karush-Kuhn-Tucker (KKT) conditions, we obtain:

$$\boldsymbol{\beta} \odot \boldsymbol{H} = \boldsymbol{0}. \tag{16}$$

Equation (16) is the fixed point equation that the solution must satisfy at convergence. According to methods in [11] and [13], the updating rule is shown as follows:

$$\boldsymbol{H} \leftarrow \boldsymbol{H} \odot \left[\frac{\hat{\boldsymbol{\Phi}} + \hat{\boldsymbol{\Theta}} + \check{\boldsymbol{\Psi}} + \check{\boldsymbol{\Pi}} + \lambda \boldsymbol{S} \boldsymbol{H} + \lambda \boldsymbol{S}^\top \boldsymbol{H}}{\check{\boldsymbol{\Phi}} + \check{\boldsymbol{\Theta}} + \hat{\boldsymbol{\Psi}} + \hat{\boldsymbol{\Pi}} + 2 \lambda \boldsymbol{D} \boldsymbol{H}} \right]^{\frac{1}{8}}, \tag{17}$$

Algorithm 1 Regularized Asymmetric Semi-NMF (RAsNMF)

Input:
Adjacency matrix $\boldsymbol{A}$, number of clusters c
Parameter:
regularization parameter λ, asymmetry parameter α
Output: $\boldsymbol{H}, \boldsymbol{Z}$

1: Initialize $\boldsymbol{H}_0, \boldsymbol{Z}_0$
2: calculate the asymmetric similarity matrix $\boldsymbol{S}$
3: calculate the degree matrix $\boldsymbol{D}$
4: **while** convergence not reached **do**
5: fix $\boldsymbol{Z}$, update $\boldsymbol{H}$ according to (17)
6: fix $\boldsymbol{H}$, update $\boldsymbol{Z}$ according to (13)
7: **end while**

where $\hat{\boldsymbol{\Lambda}}$ is a matrix that has the negative entries of matrix $\boldsymbol{\Lambda}$ replaced with 0 ($\hat{\boldsymbol{\Lambda}}_{ij} = (|\boldsymbol{\Lambda}_{ij}| + \boldsymbol{\Lambda}_{ij})/2$), and similarly $\check{\boldsymbol{\Lambda}}$ is one that has the positive entries of $\boldsymbol{\Lambda}$ replaced with 0 ($\check{\boldsymbol{\Lambda}}_{ij} = |(\boldsymbol{\Lambda}_{ij}| - \boldsymbol{\Lambda}_{ij})/2$) and

$$\begin{aligned}
&\hat{\boldsymbol{\Phi}} = \boldsymbol{A}\boldsymbol{H}\hat{\boldsymbol{Z}}^\top && ,\check{\boldsymbol{\Phi}} = \boldsymbol{A}\boldsymbol{H}\check{\boldsymbol{Z}}^\top \\
&\hat{\boldsymbol{\Theta}} = \boldsymbol{A}^\top\boldsymbol{H}\hat{\boldsymbol{Z}} && ,\check{\boldsymbol{\Theta}} = \boldsymbol{A}^\top\boldsymbol{H}\check{\boldsymbol{Z}} \\
&\check{\boldsymbol{\Psi}} = \boldsymbol{H}\check{\boldsymbol{Z}}\boldsymbol{H}^\top\boldsymbol{H}\check{\boldsymbol{Z}}^\top && ,\hat{\boldsymbol{\Psi}} = \boldsymbol{H}\hat{\boldsymbol{Z}}\boldsymbol{H}^\top\boldsymbol{H}\hat{\boldsymbol{Z}}^\top \\
&\check{\boldsymbol{\Pi}} = \boldsymbol{H}\check{\boldsymbol{Z}}^\top\boldsymbol{H}^\top\boldsymbol{H}\check{\boldsymbol{Z}} && ,\hat{\boldsymbol{\Pi}} = \boldsymbol{H}\hat{\boldsymbol{Z}}^\top\boldsymbol{H}^\top\boldsymbol{H}\hat{\boldsymbol{Z}}.
\end{aligned}$$

The details of Regularized Asymmetric Semi-NMF (RAsNMF) is summarized in Algorithm 1, and implementation is available on GitHub repository[1].

IV. Experimental Results

In this section, we perform several experiments to reach the effectiveness of the proposed algorithm in community detection in directed networks. Experiments are accomplished on real-world datasets. The proposed RAsNMF algorithm is compared to ANMF [11], RANMF [12], graph clustering based on spectral clustering [18], and normalized cut [19] . The algorithms are implemented and executed by *GNU Octave v5.2.0* on an *Intel Core i5 2520M* processor with 2.5GHz frequency.

This section addresses the effectiveness of proposed algorithm for the graph clustering. The experiments involved the comparison among clustering methods on real-world datasets. To evaluate the proposed method, efficacies of the proposed method with/without a asymmetric similarity will be analyzed. Finally, the sensitivity of parameters will also be elaborated.

A. Datasets

We utilize the *WebKB (World Wide Knowledge Base)* datasets [20] include website hyperlink information gathered from four universities of *Wisconsin*, *Texas*, *Cornell*, and *Washington*, in order to test our algorithm and also to compare with the above-mentioned methods. Nodes of the graph indicate web pages and the edges are the representative of link information between web pages. There are 5 classifications for the web pages inclusive of faculty, student, staff, course and project. Table I presents the selected datasets with the number of nodes, edges and clusters. It also shows the values of asymmetry index ai and sparsity in each graph.

[1]https://github.com/rabdollahi/RAsNMF

TABLE I
Real-world and synthetic datasets

dataset	#node	#edge	#cluster	ai	sparsity
Cornell	195	304	5	0.807	0.992
Texas	187	328	5	0.880	0.991
Wisconsin	265	530	5	0.800	0.993
Washington	230	446	5	0.853	0.993

B. Evaluation measures

In order to evaluate the clustering effectiveness of algorithms in case which the correct labels are specified, three methods of performance measurement are employed namely as NMI (Normalized Mutual Information), Jaccard similarity, and accuracy [2]. NMI is defined as an external measure for determining quality of clustering based on cluster labels and it can compare different methods of clustering with different number of clusters. If we have two cluster sets a and b, NMI is defined as

$$NMI(a,b) = \frac{I(a,b)}{\sqrt{H(a)H(b)}}, \tag{18}$$

where $I(a,b)$ is mutual information between a and b, and $H(a), H(b)$ are entropies of a and b.

Jaccard similarity evaluate the similarity by obtaining the proportion of size of the intersection to the size of their union.

$$J(a,b) = \frac{|a \cap b|}{|a \cup b|} = \frac{|a \cap b|}{|a| + |b| - |a \cap b|}. \tag{19}$$

Accuracy determines the proportion of number of true-labeled nodes to the total number of nodes in the entire network. The calculation formula of accuracy is

$$ACC(a,b) = \sum_{i=1}^{n} \frac{\delta(a_i, h(b_i))}{n}, \tag{20}$$

where $h(\cdot)$ maps each cluster label to a predicted label by the Hungarian algorithm [21] and this mapping is optimal, let $\delta(a_i, h(b_i))$ equal to 1 if $a_i = h(b_i)$ or equals to 0 otherwise.

C. Clustering Results

The implementation of the presented algorithm initiates with constructing an $n \times n$ adjacency matrix, which n is the number of nodes in the dataset. Next, we should evaluate the similarity matrix $\boldsymbol{S}$ according to the methods mentioned (cosine and asymmetric cosine similarity measures). After that we obtain the initial values of $\boldsymbol{H}$ and $\boldsymbol{Z}$ with the singular value decomposition (SVD)-based algorithm [22] and improve them using the updating formulas until the algorithm converges.

Figures 2-5 present the comparison results of graph clustering algorithms for the 4 datasets shown. The proposed algorithm is performed twice, each time using different methods of similarity

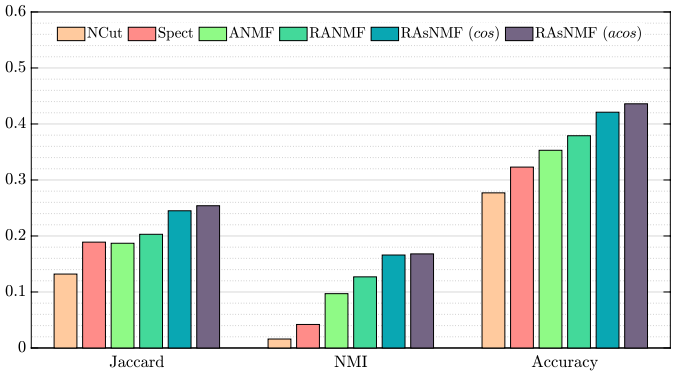

Fig. 2. Experimental results (Jaccard, Normalized Mutual Information, and Accuracy) on the *Cornell* dataset for graph clustering.

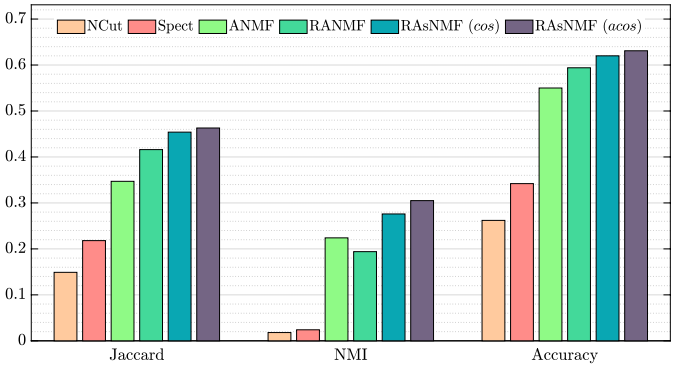

Fig. 3. Experimental results (Jaccard, Normalized Mutual Information, and Accuracy) on the *Texas* dataset for graph clustering.

which are cosine and asymmetric cosine similarities. In Figure 2 with the Cornell dataset, it seems our proposed algorithm using the cosine similarity outperforms the other methods in all three types of measurement and also asymmetric cosine similarity has even made it better in Jaccard and accuracy measures and attained the highest performance in comparison with the other algorithms. In Figure 3 using Texas dataset, the figures also show the success of our algorithms with improved outcomes in NMI for RAsNMF (acos).

In case with the Wisconsin dataset in Figure 4, the RAsNMF (cos) improved the clustering results in Jaccard similarity and NMI, but for the accuracy measure it made no difference with the RANMF outcomes. Figure 5 which displays the results in Washington dataset, tells us that both RAsNMF (cos) and RAsNMF (acos) algorithms have achieved similar performance, however in comparison with the other methods, they have been more effective. It is evident overall that asymmetric cosine similarity played a significant role in our clustering strategy and by augmenting the semi-nonnegative matrix factorization with a regularization term, the RAsNMF (acos) has gained well efficiency and remarkable improvement among the compared algorithms.

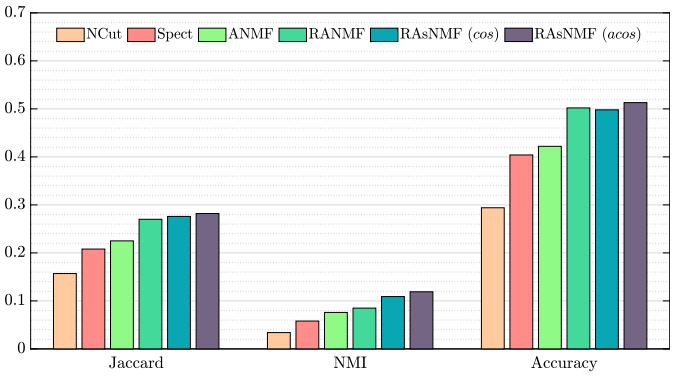

Fig. 4. Experimental results (Jaccard, Normalized Mutual Information, and Accuracy) on the *Wisconsin* dataset for graph clustering.

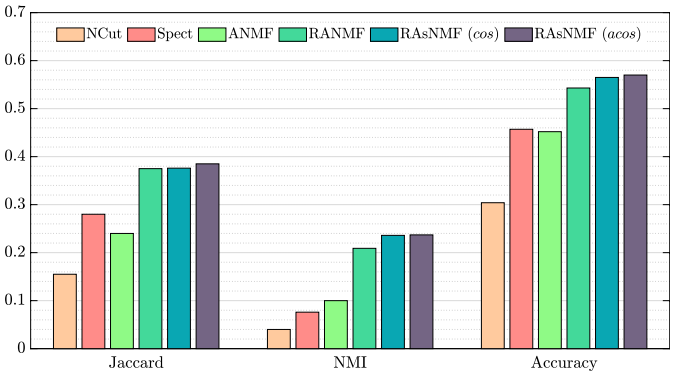

Fig. 5. Experimental results (Jaccard, Normalized Mutual Information, and Accuracy) on the *Washington* dataset for graph clustering.

D. Parameter Analysis

In this subsection, the influence of parameters on the proposed model is analyzed. RAsNMF has two parameters namely λ and α which correspond to the regularization term and the degree of asymmetry of asymmetric cosine, respectively. Thus, to make a thorough study on the two mentioned parameters we evaluated them on all datasets through a grid-search strategy and reported two results based on NMI measure. In addition, each graph consists of about 10,000 evaluations referring to 101 different λs ($\lambda \in \{0, 50, \ldots, 5000\}$) and 101 αs ($\alpha \in \{0, 0.01, \ldots, 1\}$). According to Figures 6 and 7, z-axis and colors indicate the highest and the lowest values obtained on the corresponding dataset regarding the measure which is shown using a spectrum of colors. The warm colors represent high amounts and the cold colors represent low amounts for the measure.

From Figure 6 we can see that high performance is achieved by asymmetric cosine ($\alpha < 0.47$) with small λ values, but sensitivity of λ is increased. In figure 7 is obvious that, proper α value in interval $[0.53, 0.77]$ with a more stable λ value. It concluded that asymmetric cosine similarity is more effective than cosine similarity ($\alpha = 0.5$). However, while cosine similarity is adopted, algorithm has a lower sensitivity and choosing suitable asymmetry degree is an open challenge.

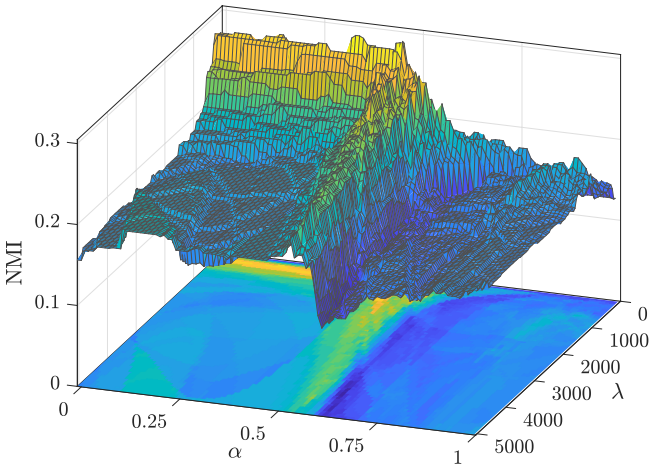

Fig. 6. Parameter analysis of *Texas* dataset.

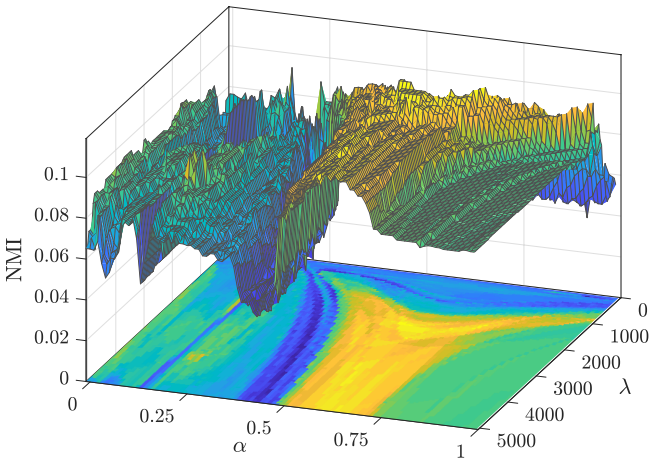

Fig. 7. Parameter analysis of *Wisconsin* dataset.

V. Conclusion

In this paper, we have proposed a novel directed graph clustering method called Asymmetric Semi-Nonnegative Matrix Factorization (AsNMF), which improves the clustering accuracy in directed graphs. Our method exploits more accurate inter-cluster dependencies in the framework of the semi-non-negative matrix factorization. The proposed method also employs an adjustable asymmetric similarity as side information to be proportional to the problem. The Algorithm has been evaluated and compared to the well-known and state-of-the-art graph clustering methods, and the obtained results show the superiority of the proposed method.

In the future, there are three directions to extend our work. (1) The distance measure of the objective function can be either distance or divergence. Kullback–Leibler divergence (relative entropy) is a distribution-wise asymmetric measure and can be a good alternative. (2) The Asymmetric Semi-NMF algorithm has the ability to use in undirected and directed signed graph clustering. (3) We can use the centrality measures such as Google PageRank in constructing a hypergraph and define an asymmetric hypergraph regularization.

References

[1] M. Newman, *Networks*. Oxford university press, 2018.

[2] C. othorel, J. D. Cruz, M. Magnani, and B. Micenkova, "Clustering attributed graphs: Models, measures and methods," *Network Science*, vol. 3, no. 3, p. 408–444, 2015.

[3] F. D. Malliaros and M. Vazirgiannis, "Clustering and community detection in directed networks: A survey," *Physics Reports*, vol. 533, no. 4, pp. 95 – 142, 2013.

[4] Y. Wang and Y. Zhang, "Nonnegative matrix factorization: A comprehensive review," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 6, pp. 1336–1353, 2013.

[5] S. A. Seyedi, P. Moradi, and F. A. Tab, "A weakly-supervised factorization method with dynamic graph embedding," in *2017 Artificial Intelligence and Signal Processing Conference (AISP)*, 2017, pp. 213–218.

[6] J. Yuan, D. Wang, and A. M. Cheriyadat, "Factorization-based texture segmentation," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 3488–3497, 2015.

[7] Z. Shajarian, S. A. Seyedi, and P. Moradi, "A clustering-based matrix factorization method to improve the accuracy of recommendation systems," in *2017 Iranian Conference on Electrical Engineering (ICEE)*, 2017, pp. 2241–2246.

[8] T. T. Vu, B. Bigot, and E. S. Chng, "Combining non-negative matrix factorization and deep neural networks for speech enhancement and automatic speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 499–503.

[9] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, Aug 2013.

[10] T. Li and C. Ding, "Nonnegative matrix factorizations for clustering: A survey." in *Data Clustering: Algorithms and Applications*, C. C. Aggarwal and C. K. Reddy, Eds. Chapman and Hall: CRC Data Mining and Knowledge Discovery Series, 2014, ch. 7, pp. 149–176.

[11] F. Wang, T. Li, X. Wang, S. Zhu, and C. Ding, "Community discovery using nonnegative matrix factorization," *Data Mining and Knowledge Discovery*, vol. 22, no. 3, pp. 493–521, 2011.

[12] A. Tosyali, J. Kim, J. Choi, and M. K. Jeong, "Regularized asymmetric nonnegative matrix factorization for clustering in directed networks," *Pattern Recognition Letters*, vol. 125, pp. 750 – 757, 2019.

[13] C. H. Q. Ding, T. Li, and M. I. Jordan, "Convex and semi-nonnegative matrix factorizations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 45–55, 2010.

[14] D. Cai, X. He, J. Han, and T. S. Huang, "Graph regularized nonnegative matrix factorization for data representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1548–1560, Aug 2011.

[15] R.-S. Wang, S. Zhang, Y. Wang, X.-S. Zhang, and L. Chen, "Clustering complex networks and biological networks by nonnegative matrix factorization with various similarity measures," *Neurocomputing*, vol. 72, no. 1-3, pp. 134–141, 2008.

[16] Y. Zhou, H. Cheng, and J. X. Yu, "Graph clustering based on structural/attribute similarities," *Proceedings of the VLDB Endowment*, vol. 2, no. 1, p. 718–729, Aug. 2009.

[17] F. Aiolli, "Efficient top-n recommendation for very large scale binary rated datasets," in *Proceedings of the 7th ACM Conference on Recommender Systems*, ser. RecSys '13. New York, NY, USA: Association for Computing Machinery, 2013, p. 273–280.

[18] S. White and P. Smyth, "A spectral clustering approach to finding communities in graphs," in *Proceedings of the 2005 SIAM international conference on data mining*. SIAM, 2005, pp. 274–285.

[19] Jianbo Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[20] M. Graven, D. DiPasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, and S. Slattery, "Learning to extract symbolic knowledge form the world wide web," in *Proceedings of 15th National Conference on Artificial Intelligence (AAAI'98)*, 1998, pp. 509–516.

[21] C. H. Papadimitriou and K. Steiglitz, *Combinatorial optimization: algorithms and complexity*. Courier Corporation, 1998.

[22] "SVD based initialization: A head start for nonnegative matrix factorization," *Pattern Recognition*, vol. 41, no. 4, pp. 1350 – 1362, 2008.