Simultaneous Attributed Network Embedding and Clustering

Anonymous authors
Paper under double-blind review

Abstract

To deal simultaneously with both, the attributed network embedding and clustering, we propose a new model. It exploits both content and structure information, capitalising on their simultaneous use. The proposed model relies on the approximation of the relaxed continuous embedding solution by the true discrete clustering one. Thereby, we show that incorporating an embedding representation provides simpler and more interpretable solutions. Experiment results demonstrate that the proposed algorithm performs better, in terms of clustering and embedding, than the state-of-art algorithms, including deep learning methods devoted to similar tasks for attributed network datasets with different proprieties.

1 Introduction

In recent years, *Attributed Networks* (AN) (Qi et al., 2012) have been used to model a large variety of real-world networks, such as academic and health care networks where both node links and attributes/features are available for analysis. Unlike plain networks in which only node links and dependencies are observed, with AN, each node is associated with a valuable set of features.

More recently, the learning representation has received a significant amount of attention as an important aim in many applications including social networks, academic citation networks and protein-protein interaction networks. Hence, *Attributed network Embedding* (ANE) (Cai et al., 2018; Yan et al., 2007; Tang et al., 2015) aims to seek a continuous low-dimensional matrix representation for nodes in a network, such that original network topological structure and node attribute proximity can be preserved in the new low-dimensional embedding.

Although, many approaches have emerged with *Network Embedding* (NE), the research on ANE (Attributed Network Embedding) is still remains to be explored (Chang et al., 2015). Unlike NE that learns from plain networks, ANE aims to capitalize both the proximity information of the network and the affinity of node attributes. Note that, due to the heterogeneity of the two information sources, it is difficult for the existing NE algorithms to be directly applied to ANE.

To sum up, the learned representation has been shown to be helpful in many learning tasks such as network clustering (Wang et al., 2017), nodes visualization (Dai et al., 2018), nodes classification (Zhu et al., 2007; Dai et al., 2018; Huang et al., 2017) and link prediction (Singh & Gordon, 2008; Pan et al., 2018). Therefore ANE is a challenging research problem due to the high-dimensionality, sparsity and non-linearity of the graph data.

2 Related Work

Frequently, embedding and clustering are used to better understand the content and structure information from the clusters. Many approaches have been proposed for the learning representation and clustering tasks (Bock, 1987; De Soete & Carroll, 1994; Vichi & Kiers, 2001; Yamamoto & Hwang, 2014) without available information from a network. With ANE we start to list some works (Pan et al., 2018; Guo et al., 2019) where the proposed algorithms show the benefits of clustering. Although existing AN clustering has been widely applied, they may achieve poor performance due to the following drawbacks:

- High risk of severe deviation of approximate continuous embedding solution from the good discrete clustering.

- Information loss among separate independent stages, i.e., continuous embedding generation and embedding discretization.

These problems result from the sequential way where the learned representation is obtained before obtaining clusters using a technical clustering. This is implicitly due to the fact that the two tasks do not aim to reach the same objective and, in addition, are carried out separately. To remedy this weakness. we propose a novel simultaneous ANE and clustering scheme which jointly; (1) learns embedding from both network and attributes information (2) learns continuous embedding and discrete clustering labels. Specifically, we explicitly enforce a discrete transformation on the intermediate continuous labels (embedding), which leads to a tractable optimization problem with a discrete solution. The key challenge is to know how to integrate the information of both node links and attributes for simultaneous node representation learning and discrete node clustering. In order to alleviate the influence caused by the information loss during the relaxation of sequential clustering, then to recover a discrete clustering solution, we use a smooth transformation (e.g., rotation) from the relaxed continuous embedding to a discrete solution. In this sense, the continuous embedding only serves as an intermediate product.

To our best knowledge, the adoption of simultaneous attributed network embedding and clustering in a unified learning framework which has not been adequately investigated yet. The goal of this work is to conduct investigations along this direction by considering matrix decomposition as the embedding framework.

The rest of the paper is organized as follows. Section 2 introduces the AN embedding problem and clustering for community detection. Section 3 provides a sound *Simultaneous Attributed Network Embedding and Clustering* (SANEC) framework for embedding and clustering. Section 4 is devoted to numerical experiments. Finally, the conclusion summarizes the advantages of our contribution.

3 PROPOSED METHOD

In this section, we descibe the Simultaneous Attributes Network Embedding and Clustering (SANEC) method. We will present the formulation of an objective function and an effective algorithm for data embedding and clustering. But first, we show how to construct two matrices $\mathbf{S}$ and $\mathbf{M}$ integrating both types of information –content and structure information– to reach our goal.

3.1 CONTENT AND STRUCTURE INFORMATION

An attributed network $G = (V, E, X)$ consists of V the set of nodes, $E \subseteq V \times V$ the set of links, and $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}2, \ldots, \mathbf{x}_n]$ where $n = |V|$ and $\mathbf{x}_i \in \mathbb{R}^d$ is the feature/attribute vector of the node v_i. Formally, the graph can be represented by two types of information, the content information $\mathbf{X} \in \mathbb{R}^{n \times d}$ and the structure information $\mathbf{A} \in \mathbb{R}^{n \times n}$, where $\mathbf{A}$ is an adjacency matrix of G and $a_{ij} = 1$ if $e_{ij} \in E$ otherwise 0; we consider that each node is a neighbor of itself, then we set $a_{ii} = 1$ for all nodes. Thereby, we model the nodes proximity by an $(n \times n)$ transition matrix $\mathbf{W}$ given by $\mathbf{W} = \mathbf{D}^{-1}\mathbf{A}$, where $\mathbf{D}$ is the degree matrix of $\mathbf{A}$ defined by $d_{ii} = \sum_{i'=1}^{n} a_{i'i}$.

In order to exploit additional information about nodes similarity from $\mathbf{X}$, we preprocessed the above dataset $\mathbf{X}$ to produce similarity graph input $\mathbf{W_X}$ of size $(n \times n)$; we construct a K-Nearest-Neighbor (KNN) graph. To this end, we use the heat kernel and L_2 distance, KNN neighborhood mode with $K = 15$ and we set the width of the neighborhood $\sigma = 1$. Note that any appropriate distance or dissimilarity measure can be used. Finally we combine in an $(n \times n)$ matrix $\mathbf{S}$, nodes proximity from both content information $\mathbf{X}$ and structure information $\mathbf{W}$. In this way, we propose to perturb the similarity $\mathbf{W}$ by adding the similarity from $\mathbf{W_X}$; we choose to take $\mathbf{S}$ defined by

$$\mathbf{S} = \mathbf{W} + \mathbf{W_X} \tag{1}$$

In Figure 3.1, we illustrate the impact of $\mathbf{W_X}$ by applying Multidimensional scaling on both $\mathbf{W}$ and $\mathbf{S}$. Note that with $\mathbf{S}$, the sparsity is overcome by the presence of $\mathbf{W_X}$. We will see later the interest of its use in $\mathbf{S}$.
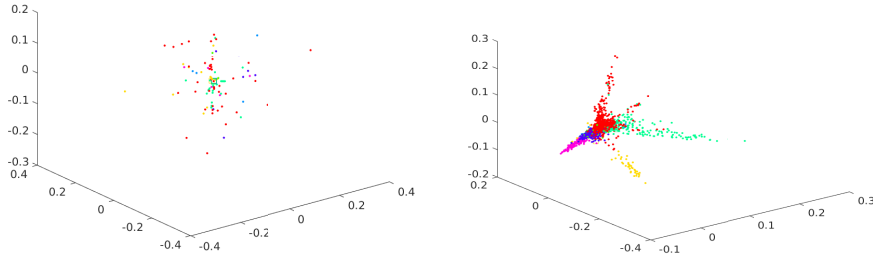
Figure 1: **MDS on W (left) and S (right): Cora dataset where W is of size** (2708×2708) **with true 7 clusters**

.

As we aim to perform clustering, we propose to integrate it in the formulation of a new data representation by assuming that nodes with the same label tend to similar social relations and similar node attributes. This idea is inspired by the fact that, the labels are strongly influenced by both content and structure information and inherently correlated to these both information sources. This reminds us that the idea behind the *Canonical Discriminant Analysis* (CDA) which is a dimension-reduction technique related to principal component analysis (PCA) and canonical correlation (Goodfellow et al., 2016). Given groups of observations with measurements on attributes, CDA derives a linear combination of the variables that has the highest possible multiple correlation with the groups. It can be viewed as a particular PCA where the observations belonging to a same group are replaced by their centroid. Thereby the new data representation referred to as $\mathbf{M} = (m_{ij})$ of size $(n \times d)$ can be considered as a multiplicative integration of both $\mathbf{W}$ and $\mathbf{X}$ by replacing each node by the centroid of their neighborhood (barycenter): i.e, $\mathbf{m}_{ij} = \sum_{k=1}^{n} \mathbf{w}_{ik}\mathbf{x}_{kj}, \forall i, j$ or

$$\mathbf{M} = \mathbf{WX}. \tag{2}$$

In Figure 3.1, it is interesting to point out the impact of $\mathbf{W}$ in the formulation of $\mathbf{M}$. We apply CDA on $\mathbf{X}$ and $\mathbf{M}$ and indicate the seven true clusters of the Cora dataset. This leads to show clusters better separated with $\mathbf{M}$ than with $\mathbf{X}$ and therefore that $\mathbf{W}$ already does a good job without clustering.
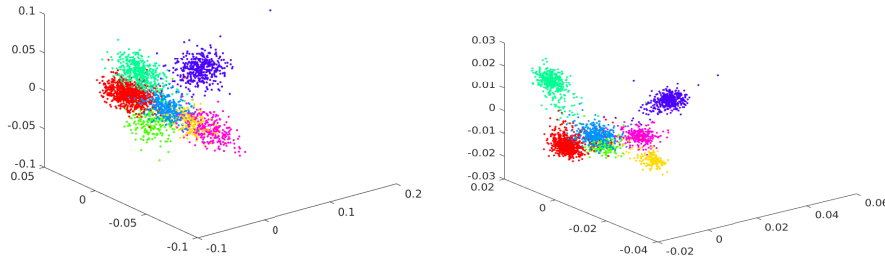

Figure 2: **CDA on X (left) and M (right): Cora dataset where X is of size** (2708×1433) **with true 7 clusters**

.

In this way, given a graph G, a graph clustering aims to partition the nodes in G into k disjoint clusters $\{C_1, C_2, ..., C_k\}$, so that: (1) nodes within the same cluster are close to each other while nodes in different clusters are distant in terms of graph structure; and (2) the nodes within the same cluster are more likely to have similar attribute values.

3.2 MODEL

Let k be the number of clusters and the number of components into which the data is embedded. With $\mathbf{M}$ and $\mathbf{S}$, the SANEC method that we propose aims to obtain the maximally informative embedding according to the clustering structure in the attributed network data. Therefore, the proposed

objective function to optimize is given by

$$\min_{\mathbf{B},\mathbf{Z},\mathbf{Q},\mathbf{G}} \left\| \mathbf{M} - \mathbf{B}\mathbf{Q}^\top \right\|^2 + \lambda \left\| \mathbf{S} - \mathbf{G}\mathbf{Z}\mathbf{B}^\top \right\|^2 \quad \mathbf{B}^\top\mathbf{B} = \mathbf{I}, \mathbf{Z}^\top\mathbf{Z} = \mathbf{I}, \mathbf{G} \in \{0,1\}^{n \times k} \tag{3}$$

where $\mathbf{G} = (g_{ij})$ of size $(n \times k)$ is a cluster membership matrix, $\mathbf{B} = (b_{ij})$ of size $(n \times k)$ is the embedding matrix and $\mathbf{Z} = (z_{ij})$ of size $(k \times k)$ is an orthonormal rotation matrix which most closely maps $\mathbf{B}$ to $\mathbf{G} \in \{0,1\}^{n \times k}$. $\mathbf{Q} \in \mathbb{R}^{d \times k}$ is the features embedding matrix. Finally, The parameter λ is a non-negative value and can be viewed as a regularization parameter.

The intuition behind the factorization of $\mathbf{M}$ and $\mathbf{S}$ is to encourage the nodes with similar proximity, those with higher similarity in both matrices, to have closer representations in the latent space given by $\mathbf{B}$. In doing so, the optimisation of (3) leads to a clustering of the nodes into k clusters given by $\mathbf{G}$. Note that, both tasks –embedding and clustering– are performed simultaneously and supported by $\mathbf{Z}$; it is the key to attaining good embedding while taking into account the clustering structure.

3.3 Optimization

To infer the latent factor matrices $\mathbf{Z}$, $\mathbf{B}$, $\mathbf{Q}$ and $\mathbf{G}$, we shall derive an alternating optimization algorithm. To this end, we rely on the following proposition.

Proposition 1 Let be $\mathbf{S} \in \mathbb{R}^{n \times n}$, $\mathbf{G} \in \{0,1\}^{n \times k}$, $\mathbf{Z} \in \mathbb{R}^{k \times k}$, $\mathbf{B} \in \mathbb{R}^{n \times k}$, we have

$$\left\| \mathbf{S} - \mathbf{G}\mathbf{Z}\mathbf{B}^\top \right\|^2 = \left\| \mathbf{S} - \mathbf{B}\mathbf{B}^\top\mathbf{S} \right\|^2 + \left\| \mathbf{S}\mathbf{B} - \mathbf{G}\mathbf{Z} \right\|^2 \tag{4}$$

Proof. We first expand the matrix norm of the left term of (4)

$$\left\| \mathbf{S} - \mathbf{G}\mathbf{Z}\mathbf{B}^T \right\|^2 = \|\mathbf{S}\|^2 + \left\| \mathbf{G}\mathbf{Z}\mathbf{B}^\top \right\|^2 - 2Tr(\mathbf{S}\mathbf{G}\mathbf{Z}\mathbf{B}^\top) \tag{5}$$

In a similar way, we obtain from the two terms of the right term of (4)

$$\begin{aligned}
\left\| \mathbf{S} - \mathbf{S}\mathbf{B}\mathbf{B}^T \right\|^2 &= \|\mathbf{S}\|^2 + \left\| \mathbf{S}\mathbf{B}\mathbf{B}^\top \right\|^2 - 2Tr(\mathbf{S}\mathbf{B}\mathbf{B}^\top\mathbf{S}^\top) \\
&= \|\mathbf{S}\|^2 + \left\| \mathbf{S}\mathbf{B}\mathbf{B}^\top \right\|^2 - 2\|\mathbf{S}\mathbf{B}\|^2 \\
&= \|\mathbf{S}\|^2 - \|\mathbf{S}\mathbf{B}\|^2 \quad \text{due to } \mathbf{B}^\top\mathbf{B} = \mathbf{I}
\end{aligned} \tag{6}$$

$$\text{and} \quad \|\mathbf{S}\mathbf{B} - \mathbf{G}\mathbf{Z}\|^2 = \|\mathbf{S}\mathbf{B}\|^2 + \|\mathbf{G}\mathbf{Z}\|^2 - 2Tr(\mathbf{S}\mathbf{B}\mathbf{Z}\mathbf{G}^\top).$$

Due also to $\mathbf{B}^\top\mathbf{B} = \mathbf{I}$, we have

$$\|\mathbf{S}\mathbf{B} - \mathbf{G}\mathbf{Z}\|^2 = \|\mathbf{S}\mathbf{B}\|^2 + \|\mathbf{G}\mathbf{Z}\mathbf{B}^\top\|^2 - 2Tr(\mathbf{S}\mathbf{G}\mathbf{Z}\mathbf{B}^\top) \tag{7}$$

Summing the two terms of (6) and (7) leads to the left term of (4).

$$\|\mathbf{S}\|^2 + \|\mathbf{G}\mathbf{Z}\|^2 - 2Tr(\mathbf{S}\mathbf{G}\mathbf{Z}\mathbf{B}^\top) = \left\| \mathbf{S} - \mathbf{G}\mathbf{Z}\mathbf{B}^T \right\|^2 \tag{8}$$

due to $\|\mathbf{G}\mathbf{Z}\|^2 = \left\| \mathbf{G}\mathbf{Z}\mathbf{B}^\top \right\|^2$. $\qquad\square$

Compute Z. Fixing $\mathbf{G}$ and $\mathbf{B}$ the problem which arises in (3) is equivalent to $\min_{\mathbf{Z}} \left\| \mathbf{S} - \mathbf{G}\mathbf{Z}\mathbf{B}^\top \right\|^2$. From proposition 1., we deduce that

$$\min_{\mathbf{Z}} \left\| \mathbf{S} - \mathbf{G}\mathbf{Z}\mathbf{B}^\top \right\|^2 \Leftrightarrow \min_{\mathbf{Z}} \left\| \mathbf{S} - \mathbf{B}\mathbf{B}^\top\mathbf{S} \right\|^2 + \left\| \mathbf{S}\mathbf{B} - \mathbf{G}\mathbf{Z} \right\|^2 \tag{9}$$

which can be reduced to $\max_{\mathbf{Z}} Tr(\mathbf{G}^\top\mathbf{S}\mathbf{B}\mathbf{Z})$ s.t. $\mathbf{Z}^\top\mathbf{Z} = \mathbf{I}$. As proved in page 29 of ten Berge (1993), let $\mathbf{U}\Sigma\mathbf{V}^\top$ be the SVD for $\mathbf{G}^\top\mathbf{S}\mathbf{B}$, then

$$\mathbf{Z} = \mathbf{U}\mathbf{V}^\top. \tag{10}$$

We can observe that this problem turns out to be similar to the well known orthogonal Procrustes problem (Schonemann, 1966).

Compute Q. Given $\mathbf{G}$, $\mathbf{Z}$ and $\mathbf{B}$, the opimization problem (3) is equivalent to $\min_{\mathbf{Q}} \left\| \mathbf{M} - \mathbf{B}\mathbf{Q}^\top \right\|^2$, and we get

$$\mathbf{Q} = \mathbf{M}^\top \mathbf{B}. \tag{11}$$

Thereby $\mathbf{Q}$ is somewhere an embedding of attributes.

Compute B. Given $\mathbf{G}$, $\mathbf{Q}$ and $\mathbf{Z}$, the problem (3) is equivalent to

$$\max_{\mathbf{B}} \; Tr((\mathbf{M}^\top \mathbf{Q} + \lambda \mathbf{S}\mathbf{G}\mathbf{Z})\mathbf{B}^\top) \quad \text{s.t.} \quad \mathbf{B}^\top \mathbf{B} = \mathbf{I}.$$

In the same manner for the computation of $\mathbf{Z}$, let $\hat{\mathbf{U}}\hat{\Sigma}\hat{\mathbf{V}}^\top$ be the SVD for $(\mathbf{M}^\top \mathbf{Q} + \lambda \mathbf{S}\mathbf{G}\mathbf{Z})$, we get

$$\mathbf{B} = \hat{\mathbf{U}}\hat{\mathbf{V}}^\top. \tag{12}$$

It is important to emphasize that, at each step, $\mathbf{B}$ exploits the information from the matrices $\mathbf{Q}$, $\mathbf{G}$, and $\mathbf{Z}$. This highlights one of the aspects of the simultaneity of embedding and clustering.

Compute G: Finally, given $\mathbf{B}$, $\mathbf{Q}$ and $\mathbf{Z}$, the problem (3) is equivalent to $\min_{\mathbf{G}} \left\| \mathbf{S}\mathbf{B} - \mathbf{G}\mathbf{Z} \right\|^2$. As $\mathbf{G}$ is a cluster membership matrix, its computation is done as follows: We fix $\mathbf{Q}$, $\mathbf{Z}$, $\mathbf{B}$. Let $\tilde{\mathbf{B}} = \mathbf{S}\mathbf{B}$ and calculate

$$g_{ik} = \begin{cases} 1 & \text{if } k = \arg\min_{k'} \left\| \tilde{\mathbf{b}}_i - \mathbf{z}_{k'} \right\|^2 \\ 0 & \text{otherwise.} \end{cases} \tag{13}$$

3.4 SANEC ALGORITHM

In summary, the steps of the SANEC algorithm relying on $\mathbf{S}$ referred to as SANEC$_\mathbf{S}$ can be deduced in Algorithm 1. The convergence of SANEC$_\mathbf{S}$ is guaranteed and depends on the initialization to reach only a local optima. Hence, we start the algorithm several times and select the best result which minimizes the objective function (3).

Algorithm 1 : SANEC$_\mathbf{S}$ algorithm

> **Input:** $\mathbf{M}$ and $\mathbf{S}$ from structure matrix $\mathbf{W}$ and content matrix $\mathbf{X}$;
> **Initialize:** $\mathbf{B}$, $\mathbf{Q}$ and $\mathbf{Z}$ with arbitrary orthonormal matrix;
> **repeat**
>   (a) - Compute $\mathbf{G}$ using (13)
>   (b) - Compute $\mathbf{B}$ using (12)
>   (c) - Compute $\mathbf{Q}$ using (11)
>   (d) - Compute $\mathbf{Z}$ using (10)
> **until** convergence
> **Output:** $\mathbf{G}$: clustering matrix, $\mathbf{Z}$: rotation matrix, $\mathbf{B}$: nodes embedding and $\mathbf{Q}$: attributes embedding

---

4 NUMERICAL EXPERIMENTS

In our work we focus on different clustering methods. In the sequel, we evaluate the SANEC algorithm with some competitive methods including recent deep learning methods.

4.1 COMPARED METHODS

We compare our method with both embedding based approaches as well as approaches directly for graph clustering. We consider classical methods and also deep learning methods; they differ in the use of available information. Some of them rely only on $\mathbf{X}$ such as K-means and others more recent on $\mathbf{X}$ and $\mathbf{W}$. Graph Encoder (Tian et al., 2014) learns graph embedding for spectral graph clustering. DNGR (Cao et al., 2016) trains a stacked denoising autoencoder for graph embedding. RTM (Chang & Blei, 2009) learns the topic distributions of each document from both text and citation. RMSC(Xia et al., 2014) employs a multi-view learning approach for graph clustering. TADW (Yang

et al., 2015) applies matrix factorization for network representation learning. `DeepWalk` (Perozzi et al., 2014) is a network representation approach which encodes social relations into a continuous embedding space. Spectral Clustering (Tang & Liu, 2011) is a widely used approach for learning social embedding. `GAE` (Kipf & Welling, 2016) is an autoencoder-based unsupervised framework for attributed network data embedding. `VGAE` (Kipf & Welling, 2016) is a variational graph autoencoder approach for graph embedding with both node links and node attributes information. `ARGA` (Pan et al., 2018) is the most recent adversarially regularized autoencoder algorithm which uses graph autoencoder to learn the embedding. `ARVGA` (Pan et al., 2018) algorithm, which uses a variational graph autoencoder to learn the embedding. For embedding based approaches, we first learn the graph embedding, and then perform `k-means` clustering algorithm based on the embedding. For our proposed method, we set the regularization parameter $\lambda \in \{0, 10^{-6}, 10^{-3}, 10^{-1}, 10^0, 10^1, 10^3\}$, and choose the best values as the final results.

4.2 DATASETS

The performances of clustering methods are evaluated using real-world datasets commonly tested with ANE where the clusters are known. Specifically, we consider three public citation network data sets, Citeseer, Cora and Wiki, which contain sparse bag-of-words feature vector for each document and a list of citation links between documents. Each document has a class label. We treat documents as nodes and the citation links as the edges. The characteristics of the used datasets are summarized in Table 1. The balance coefficient is defined as the ratio of the number of documents in the smallest class to the number of documents in the largest class while *nz* denotes the percentage of sparsity.

Table 1: Description of datasets (#: the cardinality)

datasets	# Nodes	# Attributes	# Edges	#Classes	$nz(\%)$	Balance
Cora	2708	1433	5294	7	98.73	0.22
Citeseer	3312	3703	4732	6	99.14	0.35
Wiki	2405	4973	17981	17	86.46	0.02

4.3 SENSITIVITY ANALYSIS OF λ

With the SANEC model, the parameter λ controls the role of the second term $||\mathbf{S} - \mathbf{GZB}^\top||^2$ in (3). To measure its impact on the clustering performance of SANEC$_\mathbf{S}$, we vary λ in $\{0, 10^{-6}, 10^{-3}, 10^{-1}, 10^0, 10^1, 10^3\}$. The performances in terms of *accuracy* (Acc), *normalized mutual information* (NMI) and *adjusted rand index* are depicted in Figure 4; a Acc, NMI or ARI corresponds to a better clustering result. First we note that with $\lambda = 0$ we rely only on $\min_{\mathbf{B},\mathbf{Q}} ||\mathbf{M} - \mathbf{BQ}^\top||^2$ s.t. $\mathbf{B}^\top\mathbf{B} = \mathbf{I}$. In this case, we observe poor results in terms of quality of clustering, this leads to show the impact of the second term in (3). This quality increases according to λ and we get better performance on all datasets with small values of λ. Around 10^{-2}, it is worthy to note that the clustering result is stable and less sensitive to λ. However, the performance of SANEC degrades sharply for λ greater than 10, this can be explained by the fact that the initialization of $\mathbf{G}$ which is random and can often be far from the real solution. Through, many experiments on the three datasets and others not reported here, we choose to take $\lambda = 10^{-2}$.

4.4 ATTRIBUTED NETWORK CLUSTERING

Compared to the true available clusters, in our experiments the clustering performance is assessed by ACC NMI and ARI. We repeat the experiments 50 times and the averages (mean) are reported in Table 2; the best performance for each dataset is highlighted in bold. First, we observe the high performances of methods integrating information from $\mathbf{W}$. For instance, `RTM` and `RMSC` are better than classical methods using only either $\mathbf{M}$ or $\mathbf{W}$. On the other hand, all methods including deep learning algorithms relying on $\mathbf{M}$ and $\mathbf{W}$ are better yet. However, regarding SANEC with both versions relying on $\mathbf{W}$, referred to as SANEC$_\mathbf{W}$ or $\mathbf{S}$ referred to as SANEC$_\mathbf{S}$, we note high performances for all the datasets and with SANEC$_\mathbf{S}$, we remark the impact of $\mathbf{W}_\mathbf{X}$; it learns low-dimensional representations while suits the clustering structure. To go further in our investigation and given the sparsity of $\mathbf{X}$ we proceeded to standardization tf-idf followed by L_2, as it is often used to process document-term matrices; see e.g;, (Salah & Nadif, 2017; 2019), while in the construction
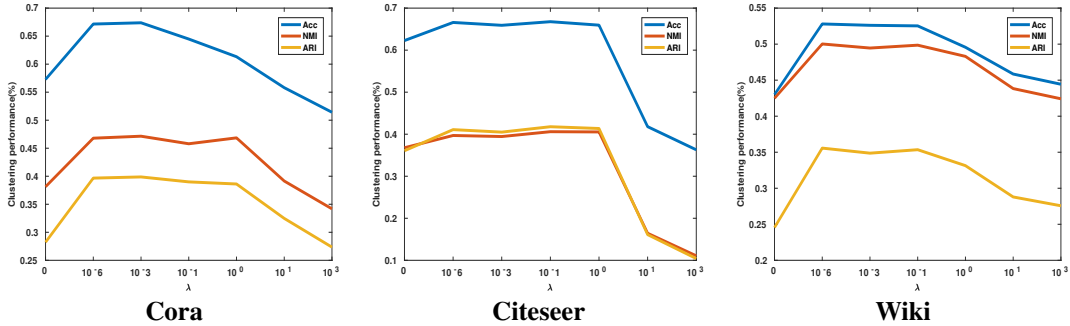
Figure 3: Sensitivity analysis of λ using `Acc`, `NMI` and `ARI`.

of $\mathbf{W_X}$ we used the cosine metric. In Figure 4 are reported the results where we observe a slight improvement.

Table 2: Clustering performances (Acc % , NMI % and ARI %) on Cora, Citeseer and Wiki datasets

Methods	Input	Datasets								
		Cora			Citeseer			Wiki		
		Acc	NMI	ARI	Acc	NMI	ARI	Acc	NMI	ARI
K-means	$\mathbf{X}$	49.22	32.10	22.96	54.01	30.54	27.86	41.72	44.02	15.07
Spectral	$\mathbf{W}$	36.72	12.67	03.11	23.89	05.57	01.00	22.04	18.17	01.46
GraphEncoder	$\mathbf{W}$	32.49	10.93	00.55	22.52	03.30	01.00	20.67	12.07	0.49
DeepWalk	$\mathbf{W}$	48.40	32.70	24.27	33.65	08.78	09.22	38.46	32.38	17.03
DNGR	$\mathbf{W}$	41.91	31.84	14.22	32.59	18.02	04.29	37.58	35.85	17.97
RTM	$\mathbf{X}, \mathbf{W}$	43.96	23.01	16.91	45.09	23.93	20.26	43.64	44.95	13.84
RMSC	$\mathbf{X}, \mathbf{W}$	40.66	25.51	08.95	29.50	13.87	04.88	39.76	41.50	11.16
TAWD	$\mathbf{X}, \mathbf{W}$	56.03	44.11	33.20	45.48	29.14	22.81	30.96	27.13	04.54
VGAE	$\mathbf{X}, \mathbf{W}$	50.20	32.92	25.47	46.70	26.05	20.56	45.09	46.76	26.34
ARGE	$\mathbf{X}, \mathbf{W}$	64.0	44.9	35.2	57.3	35.0	34.1	47.34	47.02	28.16
ARVGE	$\mathbf{X}, \mathbf{W}$	63.8	45.0	37.74	54.4	26.1	24.5	46.45	47.8	29.65
SANEC$_\mathbf{W}$	$\mathbf{X}, \mathbf{W}$	64.47	43.30	36.19	64.71	38.61	39.20	46.21	42.83	28.30
SANEC$_\mathbf{S}$	$\mathbf{X}, \mathbf{S}$	**67.38**	**47.14**	**39.88**	**66.77**	**40.60**	**41.78**	**52.80**	**50.02**	**35.57**
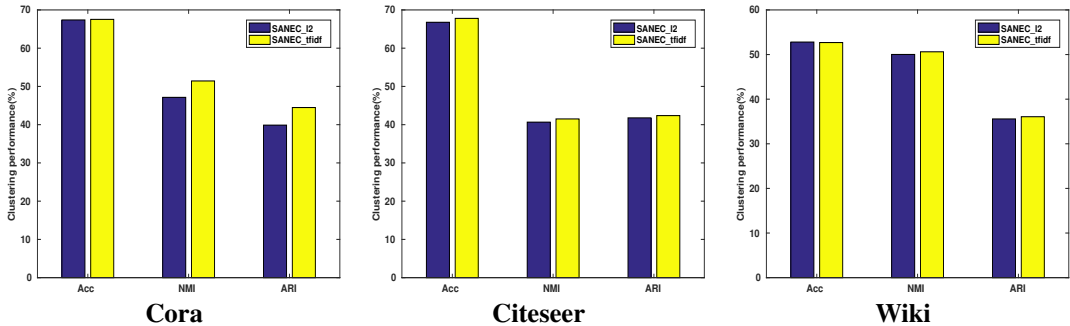
Figure 4: Evaluation of SANEC$_\mathbf{S}$ using tf-idf normalization of $\mathbf{X}$ and cosine metric for $\mathbf{W_X}$.

4.5 ATTRIBUTED NETWORK VISUALIZATION

The SANEC model, through $\mathbf{B}$, offers an embedding from which we can also observe a 2d or 3d structure into clusters. To illustrate the quality of embedding, we consider the three attributed network datasets described above and we use UMAP for data visualization (McInnes et al., 2018). The UMAP algorithm leads to dimension reduction based on manifold learning techniques and ideas from topological data analysis. As in the construction of $\mathbf{W_X}$, the number of neighbors that we have chosen with UMAP is equal to 30. Figure 4.5 McInnes et al. (2018) shows the visualization obtained on $\mathbf{X}$, on $\mathbf{M}$ and finally on $\mathbf{B}$, we easily observe the capability of SANEC$_\mathbf{S}$ to distinguish the cluster-based structures compared to visualization with the true clusters based on $\mathbf{X}$ or $\mathbf{M}$.
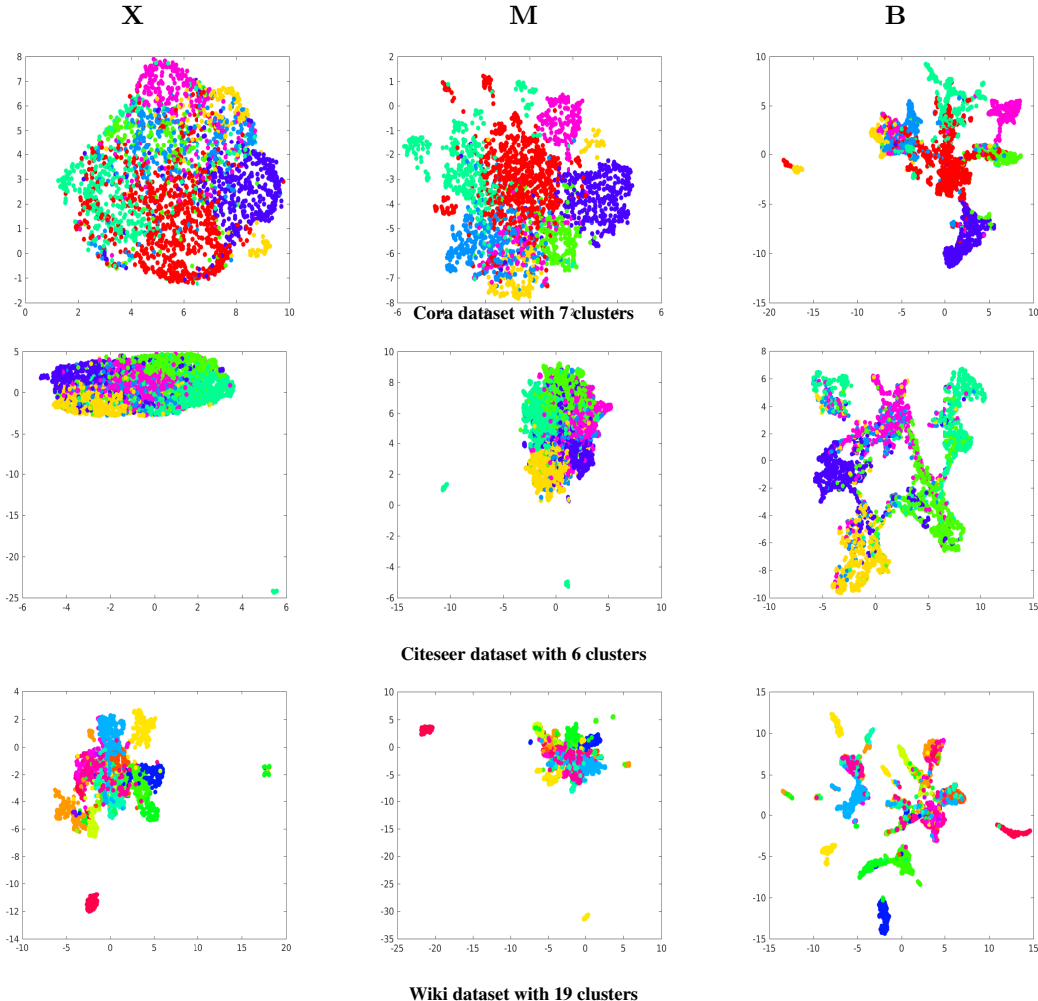
X	**M**	**B**

Cora dataset with 7 clusters

Citeseer dataset with 6 clusters

Wiki dataset with 19 clusters

Figure 5: From top to bottom and from left to right, UMAP applied on $\mathbf{X}$, $\mathbf{M}$ and $\mathbf{B}$ respectively.

5 CONCLUSION

In this paper, we proposed a novel matrix decomposition framework for simultaneous attributed network data embedding and clustering. Unlike known methods that combine the objective function of AN embedding and the objective function of clustering separately, we proposed a new single framework to perform SANEC$_\mathbf{S}$ for AN embedding and nodes clustering. We showed that the optimized objective function can be decomposed into three terms, the first is the objective function of a kind of PCA applied to $\mathbf{M}$, the second is the graph embedding criterion in a low-dimensional space, and the third is the clustering criterion. We also integrated a discrete rotation functionality, which allows a smooth transformation from the relaxed continuous embedding to a discrete solution, and guarantees a tractable optimization problem with a discrete solution. Thereby, we developed an effective algorithm capitalizing on learning representation and clustering. The obtained results show the advantages of combining both tasks over other approaches. SANEC$_\mathbf{S}$ outperforms the all recent methods devoted to the same tasks including deep learning methods which require deep models pretraining.

The proposed framework offers several perspectives and investigations. We have noted that the construction of $\mathbf{M}$ and $\mathbf{S}$ is important, it highlights the introduction of $\mathbf{W}$. As for the $\mathbf{W_X}$ we have observed that it is fundamental as it makes possible to link the information from $\mathbf{X}$ to the network; this has been verified by many experiments. Finally, as we have stressed that $\mathbf{Q}$ is an embedding of attributes, this suggests to consider also a simultaneously ANE and co-clustering.

REFERENCES

H.-H. Bock. On the interface between cluster analysis, principal component analysis, and multidimensional scaling. In *Multivariate statistical modeling and data analysis*, pp. 17–34. Springer, 1987.

HongYun Cai, Vincent W. Zheng, and Kevin Chen-Chuan Chang. A comprehensive survey of graph embedding: Problems, techniques, and applications. *IEEE Trans. Knowl. Data Eng.*, 30(9):1616–1637, 2018.

Shaosheng Cao, Wei Lu, and Qiongkai Xu. Deep neural networks for learning graph representations. 2016.

Jonathan Chang and David Blei. Relational topic models for document networks. In David van Dyk and Max Welling (eds.), *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, volume 5, pp. 81–88, 2009.

Shiyu Chang, Wei Han, Jiliang Tang, Guo-Jun Qi, Charu C. Aggarwal, and Thomas S. Huang. Heterogeneous network embedding via deep architectures. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, pp. 119–128, 2015. ISBN 978-1-4503-3664-2.

Quanyu Dai, Qiang Li, Jian Tang, and Dan Wang. Adversarial network embedding. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, 2-7, 2018*, pp. 2167–2174, 2018.

G. De Soete and JD Carroll. K-means clustering in a low-dimensional euclidean space. In *New approaches in classification and data analysis*, pp. 212–219. 1994.

Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT Press, 2016.

T. Guo, S. Pan, X. Zhu, and C. Zhang. Cfond: Consensus factorization for co-clustering networked data. *IEEE Transactions on Knowledge Data Engineering*, 31(04):706–719, 2019. ISSN 1041-4347.

Xiao Huang, Jundong Li, and Xia Hu. Label informed attributed network embedding. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, WSDM '17, pp. 731–739, 2017. ISBN 978-1-4503-4675-7.

Thomas N Kipf and Max Welling. Variational graph auto-encoders. *NIPS Workshop on Bayesian Deep Learning*, 2016.

Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

Shirui Pan, Ruiqi Hu, Guodong Long, Jing Jiang, Lina Yao, and Chengqi Zhang. Adversarially regularized graph autoencoder for graph embedding. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden.*, pp. 2609–2615, 2018.

Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, 2014.

Guo-Jun Qi, Charu C. Aggarwal, Qi Tian, Heng Ji, and Thomas S. Huang. Exploring context and content links in social media: A latent space method. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34 (5):850–862, 2012.

Aghiles Salah and Mohamed Nadif. Model-based von mises-fisher co-clustering with a conscience. In *Proceedings of the 2017 SIAM International Conference on Data Mining*, pp. 246–254. SIAM, 2017.

Aghiles Salah and Mohamed Nadif. Directional co-clustering. *Advances in Data Analysis and Classification*, 13(3):591–620, 2019.

P. Schonemann. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31 (1):1–10, 1966.

Ajit P. Singh and Geoffrey J. Gordon. Relational learning via collective matrix factorization. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, 2008.

Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. Line: Large-scale information network embedding. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15, pp. 1067–1077, 2015. ISBN 978-1-4503-3469-3.

Lei Tang and Huan Liu. Leveraging social media networks for classification. *Data Min. Knowl. Discov.*, 23(3), November 2011.

Jos MF ten Berge. *Least squares optimization in multivariate analysis*. DSWO Press, Leiden University Leiden, 1993.

Fei Tian, Bin Gao, Qing Cui, Enhong Chen, and Tie-Yan Liu. Learning deep representations for graph clustering. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI'14, 2014.

M. Vichi and H. Kiers. Factorial k-means analysis for two-way data. *Computational Statistics & Data Analysis*, 37(1):49–64, 2001.

Chun Wang, Shirui Pan, Guodong Long, Xingquan Zhu, and Jing Jiang. Mgae: Marginalized graph autoencoder for graph clustering. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, CIKM '17, pp. 889–898, 2017. ISBN 978-1-4503-4918-5.

Rongkai Xia, Yan Pan, Lei Du, and Jian Yin. Robust multi-view spectral clustering via low-rank and sparse decomposition. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI'14, 2014.

M. Yamamoto and H. Hwang. A general formulation of cluster analysis with dimension reduction and subspace separation. *Behaviormetrika*, 41(1):115–129, 2014.

Shuicheng Yan, Dong Xu, Benyu Zhang, Hong-Jiang Zhang, Qiang Yang, and Stephen Lin. Graph embedding and extensions: A general framework for dimensionality reduction. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(1), January 2007.

Cheng Yang, Zhiyuan Liu, Deli Zhao, Maosong Sun, and Edward Y. Chang. Network representation learning with rich text information. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, 2015.

Shenghuo Zhu, Kai Yu, Yun Chi, and Yihong Gong. Combining content and link for classification using matrix factorization. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, 2007.