Unsupervised Hierarchical Feature Selection on Networked Data

Yuzhe Zhang[1], Chen Chen[2], Minnan Luo[1](✉), Jundong Li[3,4], Caixia Yan[1], and Qinghua Zheng[1,5]

[1] School of Computer Science and Technology, Xi'an Jiaotong University,
Xi'an, China
{zhangyuzhe,yancaixia}@stu.xjtu.edu.cn, {minnluo,qhzheng}@xjtu.edu.cn

[2] Google Inc., Menlo Park, USA
chenannie45@gmail.com

[3] Department of Electrical and Computer Engineering, University of Virginia,
Charlottesville, USA
jundong@virginia.edu

[4] Department of Computer Science and School of Data Science,
University of Virginia, Charlottesville, USA

[5] National Engineering Lab for Big Data Analytics, Xi'an Jiaotong University,
Xi'an, China

Abstract. Networked data is commonly observed in many high-impact domains, ranging from social networks, collaboration platforms to biological systems. In such systems, the nodes are often associated with high dimensional features while remain connected to each other through pairwise interactions. Recently, various unsupervised feature selection methods have been developed to distill actionable insights from such data by finding a subset of relevant features that are highly correlated with the observed node connections. Although practically useful, those methods predominantly assume that the nodes on the network are organized in a flat structure, which is rarely the case in reality. In fact, the nodes in most, if not all, of the networks can be organized into a hierarchical structure. For example, in a collaboration network, researchers can be clustered into different research areas at the coarsest level and are further specified into different sub-areas at a finer level. Recent studies have shown that such hierarchical structure can help advance various learning problems including clustering and matrix completion. Motivated by the success, in this paper, we propose a novel unsupervised feature selection framework (HNFS) on networked data. HNFS can simultaneously learn the implicit hierarchical structure among the nodes and embed the hierarchical structure into the feature selection process. Empirical evaluations on various real-world datasets validate the superiority of our proposed framework.

Keywords: Feature selection · Attributed networks · Hierarchical structure · Pseudo labels

Y. Nah et al. (Eds.): DASFAA 2020, LNCS 12114, pp. 137–153, 2020.
https://doi.org/10.1007/978-3-030-59419-0_9

1 Introduction

Networked data is ubiquitous in many application domains. Typical examples include social networks, collaboration platforms, biological systems, and transportation networks. Normally, nodes in the above-mentioned systems are not only structurally connected, but also are associated with high-dimensional features/attributes. For example, in the biology networks, genes are connected by mutual interactions, while each of them contains numerous fragments which bring in high-dimensional features. Another representative instance is the social networks in which users are connected with each other and a diverse of user activities (e.g., posting, retweet) brings high-dimensional features. In fact, the high-dimensional data is often notoriously to tackle due to the curse of dimensionality [14]. Meanwhile, high-dimensional data not only increases the requirement of memory storage and the cost of computation, but also deteriorate the effectiveness of the algorithm due to the redundant and noisy information. To alleviate these problems, various dimensionality reduction techniques have been explored, among which feature selection has shown its effectiveness for various data mining and machine learning tasks. In particular, a feature selection algorithm can be seen as the combination of a search technique for selecting a subset of high-quality features, along with an evaluation measure which scores different subsets. The selected features would be efficient and effective to the subsequent learning tasks as the storage and computational cost is greatly reduced while the redundant and noisy information is significantly eliminated.

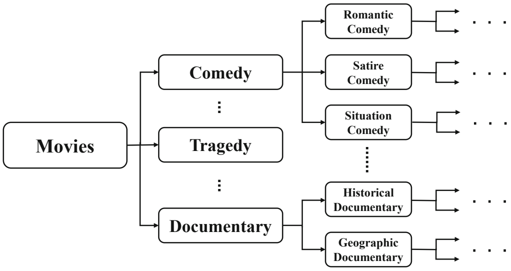


Fig. 1. Hierarchical structure of Douban movies.

Varying by the availability of labels, feature selection methods can be categorized into supervised methods and unsupervised methods. Supervised methods such as [8,25] usually gain better performance as label information is involved in the selection process. However, due to the expensive cost of amassing substantial labeled data, unsupervised feature selection has received more attention in recent years. A family of unsupervised feature selection methods employ the pseudo clustering labels of data to guide the selection phase, typical algorithms along this line include Nonnegative Discriminative Feature Selection (NDFS) [16], Robust Unsupervised Feature Selection (RUFS) [24], and Consensus Guided

Unsupervised Feature Selection (CGUFS) [18]. Although empirically effective, these cluster labels are still generated by all features which may lead to suboptimal results. Thus, some works like [15,26] chose to generate pseudo labels from external resources like connections among different data samples and has shown to be very effective. Nonetheless, these algorithms assume that the nodes on the network are organized in one-layer flat structure, which is often not the case in reality. Take the douban[1] movie rating network as an example, the movies in the platform can be classified into different genres, such as comedy, tragedy, action, *etc.* For each genre of the movies, we can further divide it into several sub-categories, which can be further divided again and again in a hierarchical manner as illustrated in Fig. 1. The data hierarchical structure has been proved to be effective in many other tasks such as representation learning [27] and recommendation [29]. Thus, it motivates us to investigate whether the success can be shifted to guide the selection of more relevant features when the label information is not available.

To address the aforementioned issues, in this paper, we propose a novel unsupervised feature selection algorithm, *i.e.*, HNFS to exploit the implicit hierarchical structure embedded on the network. Specifically, we propose to learn the implicit hierarchical structure from the network structure and measure its correlation with the node attribute information for unsupervised feature selection. The major contributions of this paper are as follow:

- Providing a principled way to learn implicit hierarchical structures of network data.
- Proposing a novel unsupervised feature selection framework which embeds the hierarchical structure learning into feature selection.
- Providing an effective alternating algorithm for the proposed algorithm.
- Demonstrating the effectiveness of the proposed framework on four commonly used real-world datasets.

2 The Proposed Framework

We first summarize the notations used throughout the paper. For a given matrix $\mathbf{A}$, $\mathbf{A}(i,j)$ denotes the (i,j)-th entry of $\mathbf{A}$. $Tr(\mathbf{A})$ denotes the trace of $\mathbf{A}$ if $\mathbf{A}$ is a square matrix. $\langle \mathbf{A}, \mathbf{B} \rangle$ equals $Tr(\mathbf{A}^T\mathbf{B})$, which means the standard inner product between two matrices. $\mathbf{I}$ is the identity matrix and $\mathbf{1}$ is a vector whose elements are all 1. For any matrix $\mathbf{A} \in \mathbb{R}^{n\times d}$, its Frobenius norm and $l_{2,1}$-norm are respectively defined as $\|\mathbf{A}\|_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{d}\mathbf{A}(i,j)^2}$ and $\|\mathbf{A}\|_{2,1} = \sum_{i=1}^{n}\sqrt{\sum_{j=1}^{d}\mathbf{A}(i,j)^2}$.

2.1 Unsupervised Feature Selection

Sparse learning has been regarded as a potent tool for feature selection [5,16]. In particular, one popular method is to embed feature selection into a clustering

[1] https://movie.douban.com/.

algorithm by selecting latent features with sparse learning [28]. Following this approach, we choose to embed feature selection into a low-rank matrix construction algorithm and apply $\ell_{2,1}$-norm on the latent representation of the original data. Let $\mathbf{X} \in \mathbb{R}^{n \times d}$ be the feature matrix which collects the feature vector of all the n nodes. Our basic model decomposes the feature matrix $\mathbf{X}$ into two matrices, *i.e.*, $\mathbf{V} \in \mathbb{R}^{n \times k}$ and $\mathbf{W} \in \mathbb{R}^{d \times k}$, and perform $\ell_{2,1}$-norm on $\mathbf{W}$ as follows:

$$\min_{\mathbf{V},\mathbf{W}} \left\| \mathbf{X} - \mathbf{V}\mathbf{W}^T \right\|_F^2 + \alpha \left\| \mathbf{W} \right\|_{2,1}, \quad s.t. \mathbf{V}^T\mathbf{V} = \mathbf{I}, \mathbf{V} \geq 0, \tag{1}$$

where $\mathbf{V}$ is the clustering indicator matrix, $\mathbf{W}$ is the latent feature matrix, and k is the number of predefined clusters. In supervised feature selection, we can regard the label information as the clustering indicator $\mathbf{V}$ to steer the selection process. But when it comes to the unsupervised situation, there is no such ground truth information, thus we choose to generate the pseudo labels by resorting to side information such as the structure information among data instances.

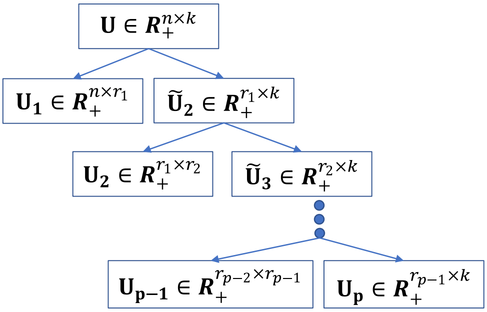

$\mathbf{U} \in R_+^{n\times k}$
$\mathbf{U_1} \in R_+^{n\times r_1}$
$\widetilde{\mathbf{U}}_2 \in R_+^{r_1\times k}$
$\mathbf{U_2} \in R_+^{r_1\times r_2}$
$\widetilde{\mathbf{U}}_3 \in R_+^{r_2\times k}$
$\mathbf{U_{p-1}} \in R_+^{r_{p-2}\times r_{p-1}}$
$\mathbf{U_p} \in R_+^{r_{p-1}\times k}$

Fig. 2. Hierarchical structure of nodes via deeply factorizing the latent feature matrix.

2.2 Latent Representation of Network Structure

Given a network G with adjacency matrix $\mathbf{A}$, we can model the latent representations of its nodes with nonnegative matrix factorization [17] as

$$\min_{\mathbf{U},\mathbf{V}} \left\| \mathbf{A} - \mathbf{U}\mathbf{V}^T \right\|_F^2, \quad s.t. \ \mathbf{U} \geq 0, \mathbf{V} \geq 0, \tag{2}$$

where $\mathbf{U} \in \mathbb{R}_+^{n \times k}$ and $\mathbf{V} \in \mathbb{R}_+^{n \times k}$. Optimizing Eq. (2) can be viewed as a clustering process over the network. Specifically, each column of $\mathbf{U}$ represents the potential definition of a community, and each row of $\mathbf{V}$ denotes the membership of a node to all k communities. Naturally, $\mathbf{U}(i,l)\mathbf{V}(l,j)$ can be regarded as the contribution of the l-th community to the edge $\mathbf{A}(i,j)$. Thus, $\widetilde{\mathbf{A}}(i,j) = \sum_{l=1}^{k} \mathbf{U}(i,l)\mathbf{V}(l,j)$ should be the result of the relationship between node i and j. Moreover, the membership of nodes obtained in $\mathbf{V}$ can act as the role of pseudo labels for unsupervised feature selection.

Through Eq. (2), we learn a one-layer representation of clustering (*i.e.*, communities in network) $\mathbf{U}$ and a community membership matrix $\mathbf{V}$). However, it assumes that the nodes on the network are organized in a one-layer flat structure, which omits the diversified and complicated organizational patterns in real-world networks as described in [34]. To learn more accurate representation of the communities on the network, we decide to further factorize the one-layer latent representation $\mathbf{U}$ to capture the implicit hierarchical structure among nodes embedded in the network. Specifically, we factorize the adjacency matrix $\mathbf{A}$ into $p+1$ nonnegative factor matrices, as follows:

$$\mathbf{A} \approx \mathbf{U}_1\mathbf{U}_2 \dots \mathbf{U}_p\mathbf{V}^T, \tag{3}$$

where $\mathbf{V} \in \mathbb{R}_+^{n\times k}$, $\mathbf{U}_i \in \mathbb{R}_+^{r_{i-1}\times r_i}(1 \leq i \leq p)$, and $n = r_0 \geq r_1 \geq \dots \geq r_{p-1} \geq r_p = k$.

Additionally, the widely used Frobenius norm for reconstruction error measuring is often very sensitive to the anomaly nodes in the network, while the $\ell_{2,1}$-norm error is often more preferred as it can enhance the robustness of the model. Hence, to collectively capture the hierarchical structures of the communities on the networks and ensure the robustness of the model, we propose to learn the latent representations of the nodes and the community assignment through the following optimization problem:

$$\begin{aligned} \min_{\mathbf{U}_i,\mathbf{V}} & \left\|\mathbf{A} - \mathbf{U}_1\mathbf{U}_2 \dots \mathbf{U}_p\mathbf{V}^T\right\|_{2,1} \\ s.t. \quad & \mathbf{V} \geq 0, \mathbf{U}_i \geq 0, i \in 1, 2, ..., p, \end{aligned} \tag{4}$$

where the original flat-structured community matrix $\mathbf{U}$ is firstly decomposed into two nonnegative matrices $\mathbf{U_1} \in \mathbb{R}_+^{n\times r_1}$ and $\widetilde{\mathbf{U_2}} \in \mathbb{R}_+^{r_1\times k}$. Following the same procedure, the latent feature matrix $\mathbf{U}$ can be further factorized into p nonnegative matrices as illustrated in Fig. 2. This formulation will lead to more accurate community membership results, *i.e.*, a better community assignment matrix $\mathbf{V}$.

2.3 The Proposed Framework – HNFS

With a hierarchy of p layers latent representations of the network, we combine Eq. (1) and Eq. (4) into a unified framework—HNFS by solving the following optimization problem:

$$\begin{aligned} \min_{\mathbf{U}_i,\mathbf{V},\mathbf{W}} & \left\|\mathbf{A} - \mathbf{U}_1\mathbf{U}_2 ... \mathbf{U}_p\mathbf{V}^T\right\|_{2,1} + \alpha \left\|\mathbf{X} - \mathbf{V}\mathbf{W}^T\right\|_F^2 + \beta \left\|\mathbf{W}\right\|_{2,1} \\ s.t. \quad & \mathbf{V}^T\mathbf{V} = \mathbf{I}, \mathbf{V} \geq 0, \mathbf{U}_i \geq 0, i \in 1, 2, ..., p, \end{aligned} \tag{5}$$

where α controls the balance between the network structure and feature information for community assignment learning; while β is a parameter to decide the sparsity of the model. With $\mathbf{W}$ fixed, the latent representation $\mathbf{V}$ is associated with both the network structure (*i.e.*, $\mathbf{A}$) and the features (*i.e.*, $\mathbf{X}$). When fixing the latent representations $\mathbf{V}$, the nodes membership learned in $\mathbf{V}$ can be viewed as the pseudo labels to guide the feature selection. As a result, the feature selection part and latent representation learning part could compliment each other and leads to a better model.

3 Optimization Algorithm

3.1 Solution

The objective function is not jointly convex *w.r.t.* all the variables, but it is convex *w.r.t.* each variable individually. Therefore, we can optimize the variables in an alternative update manner. Following [13], we propose to solve the problem with Alternating Direction Method of Multiplier (ADMM) [11]. First, we introduce two auxiliary variables $\mathbf{Z}$ and $\mathbf{E}$, and rewrite the optimization problem as:

$$\begin{aligned} \min_{\mathbf{U}_i,\mathbf{V},\mathbf{W},\mathbf{E},\mathbf{Z}} & \|\mathbf{E}\|_{2,1} + \alpha \left\|\mathbf{X} - \mathbf{V}\mathbf{W}^T\right\|_F^2 + \beta \|\mathbf{W}\|_{2,1} \\ s.t. \quad & \mathbf{Z} = \mathbf{U}_1\mathbf{U}_2...\mathbf{U}_p, \mathbf{E} = \mathbf{A} - \mathbf{Z}\mathbf{V}^T, \mathbf{V}^T\mathbf{V} = \mathbf{I} \\ & \mathbf{Z} \geq 0, \mathbf{V} \geq 0, \mathbf{U}_i \geq 0, i \in 1, 2, ..., p. \end{aligned} \tag{6}$$

The problem in Eq. (6) can be formulated as the following ADMM problem:

$$\begin{aligned} \min_{\mathbf{U}_i,\mathbf{V},\mathbf{W},\mathbf{E},\mathbf{Z}} & \|\mathbf{E}\|_{2,1} + \alpha \left\|\mathbf{X} - \mathbf{V}\mathbf{W}^T\right\|_F^2 + \beta \|\mathbf{W}\|_{2,1} \\ & + \langle \mathbf{Y}_1, \mathbf{Z} - \mathbf{U}_1\mathbf{U}_2...\mathbf{U}_p \rangle + \langle \mathbf{Y}_2, \mathbf{A} - \mathbf{Z}\mathbf{V}^T - \mathbf{E} \rangle \\ & + \frac{\mu}{2}(\|\mathbf{Z} - \mathbf{U}_1\mathbf{U}_2...\mathbf{U}_p\|_F^2 + \left\|\mathbf{A} - \mathbf{Z}\mathbf{V}^T - \mathbf{E}\right\|_F^2) \\ & s.t. \mathbf{V}^T\mathbf{V} = \mathbf{I}, \mathbf{Z} \geq 0, \mathbf{V} \geq 0, \mathbf{U}_i \geq 0, i \in 1, 2, \ ..., p, \end{aligned} \tag{7}$$

where $\mathbf{Y}_1, \mathbf{Y}_2$ are two Lagrangian multipliers, and μ is a scalar to control the penalty for the violation of equality constraints (*i.e.*, $\mathbf{Z} = \mathbf{U}_1\mathbf{U}_2 \ldots \mathbf{U}_p$ and $\mathbf{E} = \mathbf{A} - \mathbf{Z}\mathbf{V}^T$).

Update E. Fixing all other variables except $\mathbf{E}$, the objective function can be reformulated as:

$$\min_{\mathbf{E}} \frac{1}{2} \left\|\mathbf{E} - (\mathbf{A} - \mathbf{Z}\mathbf{V}^T + \frac{1}{\mu}\mathbf{Y}_2)\right\|_F^2 + \frac{1}{\mu} \|\mathbf{E}\|_{2,1}. \tag{8}$$

The equation has a closed-form solution by using the following Lemma [19].

Lemma 1. Let $\mathbf{Q} = [\mathbf{q}_1; \mathbf{q}_2; \ldots; \mathbf{q}_m]$ be a given matrix and λ be a positive scalar. If the optimal solution of

$$\min_{\mathbf{W}} \frac{1}{2} \|\mathbf{W} - \mathbf{Q}\|_F^2 + \lambda \|\mathbf{W}\|_{2,1} \tag{9}$$

is $\mathbf{W}^*$, then the ith row of $\mathbf{W}^*$ is

$$\mathbf{w}_i^* = \begin{cases} (1 - \frac{\lambda}{\|\mathbf{q}_i\|})\mathbf{q}_i \; if \; \|\mathbf{q}_i\| > \lambda \\ 0 \qquad\qquad\qquad otherwise. \end{cases} \tag{10}$$

Suppose $\mathbf{Q} = \mathbf{A} - \mathbf{Z}\mathbf{V}^T + \frac{1}{\mu}\mathbf{Y}_2$, $\mathbf{E}$ can be updated as follow by using Lemma 1:

$$\mathbf{e}_i = \begin{cases} (1 - \frac{1}{\mu\|\mathbf{q}_i\|})\mathbf{q}_i \; if \; \|\mathbf{q}_i\| > \frac{1}{\mu} \\ 0 \qquad\qquad\qquad otherwise. \end{cases} \tag{11}$$

Update V. We follow the same strategy in [23] to update $\mathbf{V}$. Note that the constraints of $\mathbf{V}$ are the same in [23] and ours. Removing irrelevant terms to $\mathbf{V}$ from Eq. (7), the optimization problem can be rewritten as:

$$\min_{\mathbf{V}^T\mathbf{V}=\mathbf{I}} \frac{\mu}{2} \left\| \mathbf{A} - \mathbf{Z}\mathbf{V}^T - \mathbf{E} + \frac{1}{\mu}\mathbf{Y}_2 \right\|_F^2 + \alpha \left\| \mathbf{X} - \mathbf{V}\mathbf{W}^T \right\|_F^2 . \tag{12}$$

After expanding the objective function and dropping terms that are independent of $\mathbf{V}$, we get

$$\min_{\mathbf{V}^T\mathbf{V}=\mathbf{I}} \frac{\mu}{2} \|\mathbf{V}\|_F^2 - \mu\langle \mathbf{N}, \mathbf{V} \rangle, \tag{13}$$

where $\mathbf{N} = (\mathbf{A}^T - \mathbf{E}^T + \frac{1}{\mu}\mathbf{Y}_2^T)\mathbf{Z} - \frac{2\alpha}{\mu}\mathbf{X}\mathbf{W}$. The above equation can be further simplified to a more compact form as $\min\limits_{\mathbf{V}^T\mathbf{V}=\mathbf{I}} \|\mathbf{V} - \mathbf{N}\|_F^2$. According to [13], $\mathbf{V}$ can be updated by the following equation in which $\mathbf{P}$ and $\mathbf{Q}$ are left and right singular values of the SVD decomposition of $\mathbf{N}$:

$$\mathbf{V} = \mathbf{P}\mathbf{Q}^T. \tag{14}$$

Update W. The update rule for $\mathbf{W}$ is similar as $\mathbf{E}$. When other variables except $\mathbf{W}$ are fixed and terms that are irrelevant to $\mathbf{W}$ are removed, the optimization problem for $\mathbf{W}$ can be rewritten as:

$$\min_{\mathbf{W}} \alpha \left\| \mathbf{X} - \mathbf{V}\mathbf{W}^T \right\|_F^2 + \beta \|\mathbf{W}\|_{2,1} . \tag{15}$$

Using the fact that $\mathbf{V}^T\mathbf{V} = \mathbf{I}$, it can be reformulated as

$$\min_{\mathbf{W}} \frac{1}{2} \left\| \mathbf{W} - \mathbf{V}\mathbf{X}^T \right\|_F^2 + \frac{\beta}{2\alpha} \|\mathbf{W}\|_{2,1} . \tag{16}$$

Again, the above equation has a closed-form solution according to Lemma 1. Let $\mathbf{K} = \mathbf{V}\mathbf{X}^T$, then

$$\mathbf{w}_i = \begin{cases} (1 - \frac{\beta}{2\alpha\|\mathbf{k}_i\|})\mathbf{k}_i \; if \; \|\mathbf{k}_i\| > \frac{\beta}{2\alpha} \\ 0 \qquad\qquad\qquad otherwise. \end{cases} \tag{17}$$

Algorithm 1. Algorithm 1 The Proposed HNFS algorithm

```
Input: The data matrix $\mathbf{X}$ and the adjacency matrix $\mathbf{A}$
       The layer size of each layer $\mathbf{r}_i$
       The regularization parameter $\alpha$, $\beta$
       The number of selected features m
Output: The most $m$ relevant features.
Initialize $\mu = 10^{-3}$, $\rho = 1.1$, $\mathbf{U}_i = 0$, $\mathbf{V} = 0$ ( or initialized using K-means)
while not convergence do
    Calculate $\mathbf{Q} = \mathbf{A} - \mathbf{Z}\mathbf{V}^T + \frac{1}{\mu}\mathbf{Y}_2$
    Update $\mathbf{E}$ by Eq. (11)
    Calculate $\mathbf{K} = \mathbf{V}\mathbf{X}^T$
    Update $\mathbf{W}$ by Eq. (17)
    Calculate $\mathbf{T} = \frac{1}{2}[(\mathbf{A} - \mathbf{E} + \frac{1}{\mu}\mathbf{Y}_2)\mathbf{V} + \mathbf{U} + \frac{1}{\mu}\mathbf{Y}_1]$
    Update $\mathbf{Z}$ by Eq. (19)
    Calculate $\mathbf{S}_i = (\mathbf{H}_i^T\mathbf{H}_i)^{-1}\mathbf{H}_i^T(\mathbf{Z} - \frac{\mathbf{Y}_1}{\mu})\mathbf{B}_i^T(\mathbf{B}_i\mathbf{B}_i^T)^{-1}$
    Update $\mathbf{U}_i$ by Eq. (24)
    Calculate $\mathbf{N} = (\mathbf{A}^T - \mathbf{E}^T + \frac{1}{\mu}\mathbf{Y}_2^T)\mathbf{Z} - \frac{2\alpha}{\mu}\mathbf{X}\mathbf{W}$
    Update $\mathbf{V}$ by Eq. (14)
    Update $\mathbf{Y}_1, \mathbf{Y}_2$ and $\mu$ by Eq. (25), Eq. (26) and Eq. (27)
end while
Sort each feature of $\mathbf{X}$ according to $\|\mathbf{w}_i\|_2$ in descending order and select the top-m
features
```

Update Z. By removing other irrelevant parts to $\mathbf{Z}$, the objective function can be rewritten as:

$$\min_{\mathbf{Z}\geq 0} \frac{\mu}{2}\left\|\mathbf{A} - \mathbf{Z}\mathbf{V}^T - \mathbf{E} + \frac{1}{\mu}\mathbf{Y}_2\right\|_F^2 + \frac{\mu}{2}\left\|\mathbf{U}_1\mathbf{U}_2 \dots \mathbf{U}_p - \mathbf{Z} + \frac{1}{\mu}\mathbf{Y}_1\right\|_F^2. \tag{18}$$

By setting the derivative of Eq. (18) *w.r.t.* $\mathbf{Z}$ to zero, we get $2\mathbf{Z} = (\mathbf{A} - \mathbf{E} + \frac{1}{\mu}\mathbf{Y}_2)\mathbf{V} + \mathbf{U} + \frac{1}{\mu}\mathbf{Y}_1$. Let $\mathbf{T} = \frac{1}{2}[(\mathbf{A} - \mathbf{E} + \frac{1}{\mu}\mathbf{Y}_2)\mathbf{V} + \mathbf{U} + \frac{1}{\mu}\mathbf{Y}_1]$. Then $\mathbf{Z}$ can be updated as:

$$\mathbf{Z}_{i,j} = max(\mathbf{T}_{i,j}, 0). \tag{19}$$

Update $\mathbf{U}_i$. By fixing all the variables except $\mathbf{U}_i$, the objective function in Eq. (7) is reduced to:

$$\min_{\mathbf{U}_i\geq 0} \frac{\mu}{2}\left\|\mathbf{H}_i\mathbf{U}_i\mathbf{B}_i - \mathbf{Z} + \frac{1}{\mu}\mathbf{Y}_1\right\|_F^2, \tag{20}$$

where $\mathbf{H}_i$ and $\mathbf{B}_i$, $1 \leq i \leq p$, are defined as:

$$\mathbf{H}_i = \begin{cases} \mathbf{U}_1\mathbf{U}_2 \dots \mathbf{U}_{i-1} & if \quad i \neq 1 \\ \mathbf{I} & if \quad i = 1, \end{cases} \tag{21}$$

and

$$\mathbf{B}_i = \begin{cases} \mathbf{U}_{i+1}\mathbf{U}_{i+2} \dots \mathbf{U}_p & if \quad i \neq p \\ \mathbf{I} & if \quad i = p. \end{cases} \tag{22}$$

By setting the derivative of Eq. (20) *w.r.t.* $\mathbf{U}_i$ to zero, we get

$$\mathbf{H}_i^T\mathbf{H}_i\mathbf{U}_i\mathbf{B}_i\mathbf{B}_i^T - \mathbf{H}_i^T(\mathbf{Z} - \frac{\mathbf{Y}_1}{\mu})\mathbf{B}_i^T = 0, \tag{23}$$

where $\mathbf{H}_i^T\mathbf{H}_i$ is a positive semi-definite matrix, the same is true of $\mathbf{B}_i\mathbf{B}_i^T$. Let $\mathbf{S}_i = (\mathbf{H}_i^T\mathbf{H}_i)^{-1}\mathbf{H}_i^T(\mathbf{Z} - \frac{\mathbf{Y}_1}{\mu})\mathbf{B}_i^T(\mathbf{B}_i\mathbf{B}_i^T)^{-1}$, then $\mathbf{U}_i$ has a closed-form solution:

$$\mathbf{U}_i(j,k) = max(\mathbf{S}_i(j,k), 0). \tag{24}$$

Update $\mathbf{Y}_1, \mathbf{Y}_2$ and μ. After updating the variables, the ADMM parameters should be updated. According to [4], they can be updated as follows:

$$\mathbf{Y}_1 = \mathbf{Y}_1 + \mu(\mathbf{Z} - \mathbf{U}_1\mathbf{U}_2 \dots \mathbf{U}_p), \tag{25}$$

$$\mathbf{Y}_2 = \mathbf{Y}_2 + \mu(\mathbf{A} - \mathbf{Z}\mathbf{V}^T - \mathbf{E}), \tag{26}$$

$$\mu = \rho\mu. \tag{27}$$

Here, $\rho > 1$ is a parameter to control the convergence speed. The larger ρ is, the fewer iterations we require to get the convergence, while the precision of the final objective function value may be sacrificed. In Algorithm 1, we summarize the procedures for optimizing Eq. (6).

Table 1. Detailed information of the datasets.

	Wiki	BlogCatalog	Flickr	DBLP
#Users	2,405	5,196	7,575	18,448
#Features	4,937	8,189	12,047	2,476
#Links	17,981	171,743	239,738	45,611
#Classes	19	6	9	4

4 Experiments

4.1 Experimental Settings

Datasets. The experiments are conducted on four commonly used real-world networks datasets, including Wiki[2], BlogCatalog[3], Flickr (See footnote 3) and DBLP[4]. The detailed statistics of these datasets are listed in Table 1.

[2] https://github.com/thunlp/OpenNE/tree/master/data/wiki.

[3] http://dmml.asu.edu/users/xufei/datasets.html.

[4] https://www.aminer.cn/citation.

- **Wiki:** Wiki is a document network which is composed of hyperlinks between wikipedia documents. Each document is displayed by a high-dimensional vector which indicates the word frequency count of itself. These documents are classified into dozens of predefined classes.
- **BlogCatalog:** BlogCatalog is a social blog directory in which users can register their blogs under different predefined categories [31]. Names, ids, blogs, the associated tags and blog categories form the content information while the class label is selected from a predefined list of categories, indicating the interests of each user.
- **Flickr:** Flickr is a content sharing platforms, with a focus on photos, where users can share their contents, upload tags and subscribe to different interest groups [32]. Besides, users interact with others forming link information while groups that users joined can be treated as class labels.
- **DBLP:** DBLP is a part of the DBLP bibliographic network dataset. It contains papers from four research areas: Database, Data Mining, Artificial Intelligence and Computer Vision. Each paper's binary feature vectors indicate the presence/absence of the corresponding word in its title.

Baseline Methods. We compare our proposed framework HNFS with the following seven unsupervised feature selection algorithms, which can be divided into two groups. The first five algorithms only consider the attribute information while the latter two take both attribute information and structure information into consideration. Following are the comparing methods used in our experiment.

- **LapScore:** Laplacian Score is a filter method for feature selection which is independent to any learning algorithm [2]. The importance of a feature is evaluated by its power of locality preserving, or, Laplacian Score [12].
- **RUFS:** RUFS is a robust unsupervised feature selection approach where robust label learning and robust feature learning are simultaneously performed via orthogonal nonnegative matrix factorization and joint $\ell_{2,1}$-norm minimization [24].
- **UDFS:** UDFS incorporates discriminative analysis and $\ell_{2,1}$-norm minimization into a joint framework for unsupervised feature selection under the assumption that the class label of input data can be predicted by a linear classifier. [33].
- **GreedyFS**: GreedyFS is an effective filter method for unsupervised feature selection which first defines a novel criterion that measures the reconstruction error of the selected data and then selects features in a greedy manner based on the proposed criterion [10].
- **MCFS:** By using spectral regression [6] with $\ell_{2,1}$-norm regularization, MCFS suggests a principled way to measure the correlations between different features without label information. Thus, MCFS can well handle the data with multiple cluster structure [5].
- **NetFS:** NetFS is an unsupervised feature selection framework for networked data, which embeds the latent representation learning into feature selection [15].

- **HNFS-flat:** HNFS-flat is a variant of our proposed framework which only considers the flat structure of networks by setting p = 1 in our model.

Metrics and Settings. Following the standard ways to assess unsupervised feature selection, we evaluate different feature selection algorithms by evaluating the clustering performance with the selected features. Two commonly adopted clustering performance metrics [14] are used: (1) *normalized mutual information* (NMI) and (2) *accuracy* (ACC). The parameter settings of the baseline methods all follow the suggestions by the original papers [5,12,33]. For our proposed method, we tune the model parameters by a "grid-search" strategy from {0.001,0.01,0.1,1,10,100,1000} and the best clustering results are reported. We implement HNFS with the number of layers $p = 2$. Although different layers $p \in \{2, 3, 4, 5, 6\}$ are tried, the performance improvement is not significant while more running time is required. Meanwhile, we specifiy the size of layer $r_1 = 256$ and we will explain the reason later. In the experiments, each feature selection algorithm is first used to select a certain number of features, then we use K-means to cluster nodes into different clusters based on the selected features. Since K-means may converge to local optima, we repeat the experiments 20 times and report the average results.

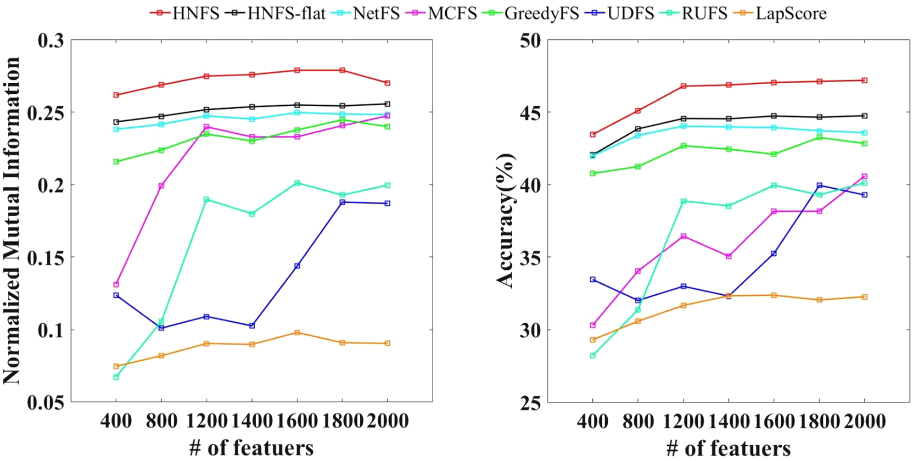


Fig. 3. Clustering results with different feature selection algorithms on Blogcatalog dataset.

4.2 Quality of Selected Features

In this subsection, we compare the quality of the selected features by our model and other baseline methods on all the datasets. The number of selected features varies from $\{400, 800, 1200, 1400, 1600, 1800, 2000\}$. The results are shown

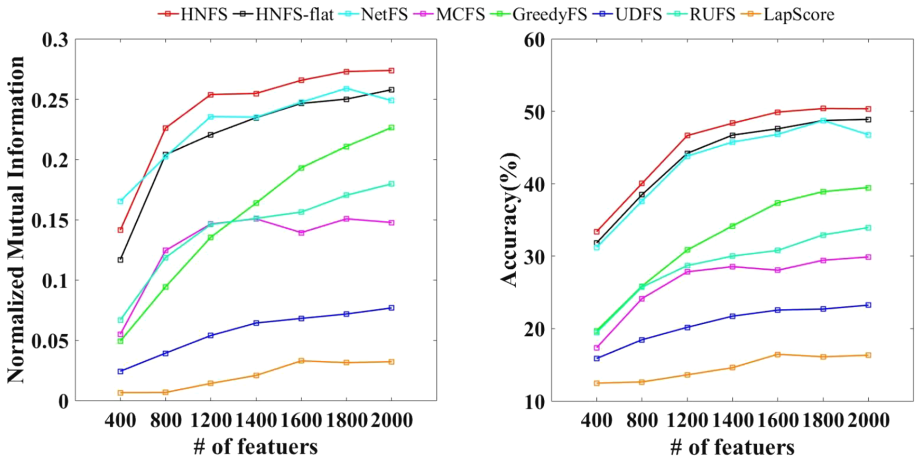


Fig. 4. Clustering results with different feature selection algorithms on Flickr dataset.

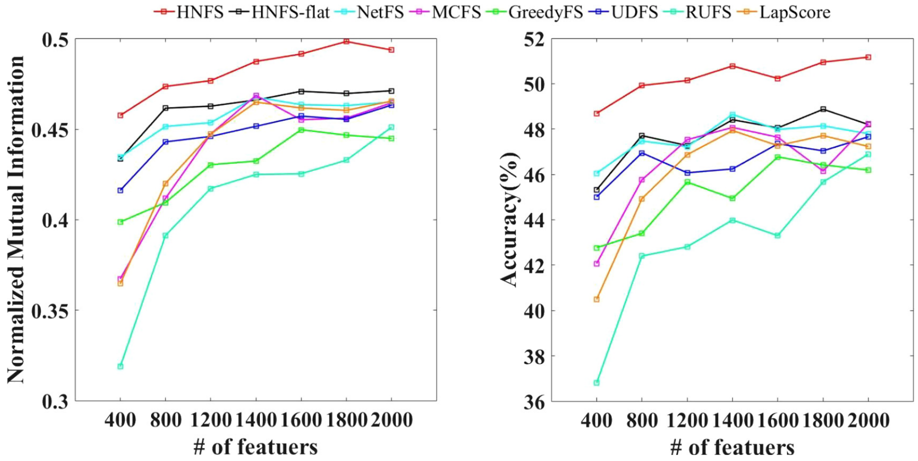


Fig. 5. Clustering results with different feature selection algorithms on Wiki dataset.

in Fig. 3, 4, 5 and 6. The higher the ACC and NMI values are, the better the feature selection performance is. We have the following observations based on the experimental results:

- The methods that consider both attribute information and structure information obtain much better results than the ones that only exploit feature information especially in Blogcatalog and Flickr. It is because that these two datasets contain abundant structure information while Wiki and DBLP have sparse adjacency matrix. Despite of this, methods that consider structure information can still benefit from it in latter two datasets in most situations.

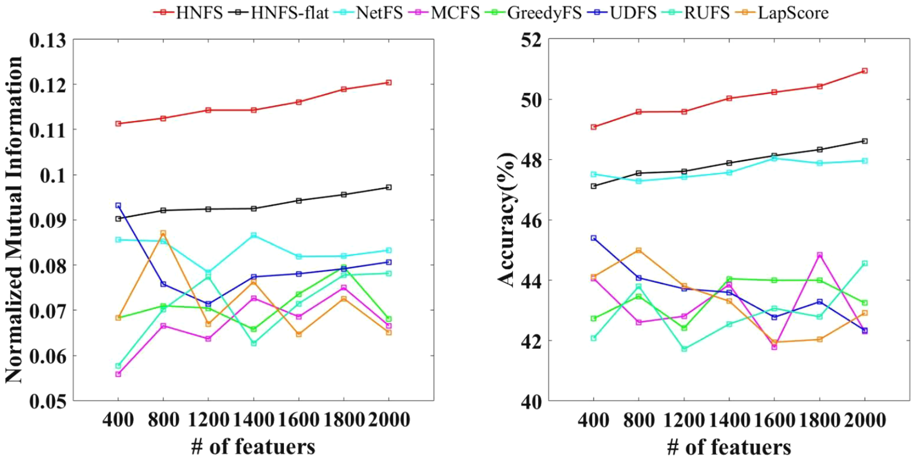
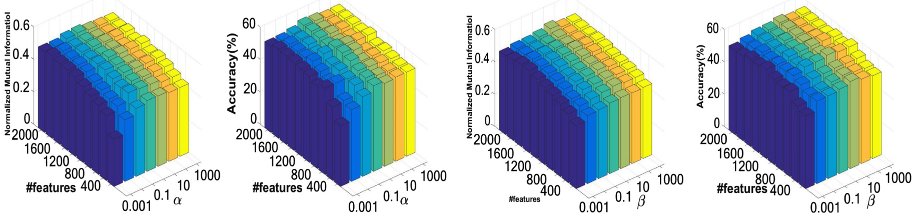


Fig. 6. Clustering results with different feature selection algorithms on DBLP dataset.

It implies that when the label information is not explicitly given, network structure indeed can help us select more relevant features.

- HNFS and NetFS consider the network structure differently. NetFS regard it as a flat-structure while real-world networks usually exhibit hierarchical structures which should be fully considered. The results between our model and NetFS prove that implicit hierarchical structures of networks can improve the performance of feature selection. The observations are further confirmed by the improvement of HNFS over its flat-structure variant HNFS-flat.
- In Wiki and DBLP datasets, HNFS performs well with only a few hundred of features. BlogCatalog and Flickr have more features than the first two, but HNFS still obtains good clustering performance with only around 1/10 and 1/20 of total features, respectively.



The effect of parameter α. The effect of parameter β.

Fig. 7. Parameter analysis on Wiki.

4.3 Parameter Analysis

Our model has two regularization parameters α and β. α controls the balance between the network structure and feature information for feature selection, while β determines the sparsity of the model. To discuss the influences of these two parameters, we choose to fix one parameter each time and change the other one to see how the clustering results change. Due to space limit, we only report the results on Wiki in Fig. 7. We first make the parameter β equal to 10 and vary the parameter α as $\{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$. We can see from Fig. 7 that when α is around 10 we can get a relatively better clustering performance. Then we make α equal to 1 and vary the parameter β as $\{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$. As shown in Fig. 7, with the increase of β, the clustering performance first increases then becomes stable. The reason is that a small α will reduce the feature sparsity of the model, which is not of great benefit to feature selection. Besides, the experimental results show that the clustering performance is more sensitive to the number of selected features compared with α and β. However, picking the proper number of features is still an open problem that requires deep investigation.

5 Related Work

5.1 Traditional Feature Selection

Depending on the existence of the label information, feature selection algorithms can be broadly divided into supervised and unsupervised methods. Supervised feature selection algorithms assess feature relevance via its correlation with the class labels [22,35]. According to the adopted strategies, we can further divide supervised feature selection into filter methods and wrapper methods [14]. Filter methods pay attention to feature selection part which means they are independent of any learning algorithms. On the contrary, wrapper methods have a close relationship with the learning algorithm. They use the learning performance to access the quality of selected features iteratively, which is often computationally expensive. Unsupervised feature selection algorithms, on the other hand, have attracted a surge of research attention due to its effectiveness in addressing unlabeled data [1,3,33]. Without label information to access the importance of features, unsupervised feature selection methods [7,30] need some alternative criteria to decide which features to select, such as data reconstruction error [9], local discriminative information [16,33], and data similarity [12,36]. To effectively select a subset of features, sparsity regularizations like l_1-norm and $\ell_{2,1}$-norm [5,15,16,33] have been extensively used in unsupervised feature selection.

5.2 Unsupervised Feature Selection with Pseudo Labels

Furthermore, to compensate the shortage of labels, many unsupervised feature selection methods tend to explore some other information among data instances to guide the feature selection procedure, namely pseudo labels. The result of clustering has been commonly used as pseudo labels in many unsupervised feature selection works. For example, NDFS [16] combines the result of spectral clustering with the traditional feature selection and obtain better performance. EUFS [28] and RUFS [24] utilize the result of spectral clustering in the same way and only change the part of feature selection to make it more robust. Since the spectral clustering can help to select feature, so do other clustering methods. CGUFS [18] proposes to learn a consensus clustering results from multitudinous clustering algorithms, which leads to better clustering accuracy with high robustness. But, the clustering result obtained by high-dimensional feature matrix may contain numerous noise. Thus, some other methods attempt to utilize structure information as the pseudo labels, namely the adjacency matrix. LUFS [26] first extract social dimensions and then utilize them for selecting discriminative features on the attributed networks while NetFS [15] embeds the latent representation obtained from structure information into feature selection. However, these works are substantially different from our proposed framework HNFS as they omit the hierarchical structure among data instances. The hierarchical information has demonstrated its importance in supervised feature selection [20,21], which facilitates the investigation of HNFS in this paper. Besides, HNFS provides an iterative way to learn the implicit hierarchical structures and feature importance measures simultaneously and the feature selection part becomes more robust compared with other unsupervised feature selection algorithms.

6 Conclusion and Future Work

In this paper, we propose an unsupervised feature selection framework HNFS for networked data. Specifically, the proposed method can effectively capture the implicit hierarchical structure of the network while measuring its correlation with node attributes for feature selection. Methodologically, we perform Alternating Direction Method of Multiplier (ADMM) to optimize the objective function. Extensive experimental results on four real-world network datasets have validated the effectiveness of our model.

There are several directions worth further investigation. First, it would be meaningful to study the effectiveness of other hierarchical network representation methods in contrast to the nonnegative matrix factorization method used in this work. Second, real-world networks are evolving over time, which means both the network structure and the features are changing timely. Thus how to generalize the proposed method in a dynamic setting would be another interesting research direction.

Acknowledgement. This work was supported by National Nature Science Foundation of China (No. 61872287, No. 61532015, and No. 61872446), Innovative Research Group of the National Natural Science Foundation of China (No. 61721002), Innovation Research Team of Ministry of Education (IRT_17R86), and Project of China Knowledge Center for Engineering Science and Technology. Besides, this research was funded by National Science and Technology Major Project of the Ministry of Science and Technology of China (No. 2018AAA0102900).

References

1. Alelyani, S., Tang, J., Liu, H.: Feature selection for clustering: a review. Data Cluster.: Algorithms Appl. **21**, 110–121 (2013)
2. Belkin, M., Niyogi, P.: Laplacian eigenmaps and spectral techniques for embedding and clustering. In: NIPS (2001)
3. Boutsidis, C., Mahoney, M.W., Drineas, P.: Unsupervised feature selection for the k-means clustering problem. In: NIPS (2009)
4. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations & Trends in Machine Learning **3**, 1–122 (2011)
5. Cai, D., He, X.: Unsupervised feature selection for multi-cluster data. In: KDD (2010)
6. Cai, D., He, X., Han, J.: Spectral regression for efficient regularized subspace learning. In: ICCV (2007)
7. Dong, X., Zhu, L., Song, X., Li, J., Cheng, Z.: Adaptive collaborative similarity learning for unsupervised multi-view feature selection. In: IJCAI (2018)
8. Fan, M., Chang, X., Zhang, X., Wang, D., Du, L.: Top-k supervise feature selection via ADMM for integer programming. In: IJCAI (2017)
9. Farahat, A.K., Ghodsi, A., Kamel, M.S.: An efficient greedy method for unsupervised feature selection. In: ICDM (2011)
10. Farahat, A.K., Ghodsi, A., Kamel, M.S.: Efficient greedy feature selection for unsupervised learning. Knowl. Inf. Syst. **2**, 285–310 (2013). https://doi.org/10.1007/s10115-012-0538-1
11. Gabay, D.: A dual algorithm for the solution of nonlinear variational problems via finite element approximation. Comput. Math. Appl. **2**, 17–40 (1976)
12. He, X., Cai, D., Niyogi, P.: Laplacian score for feature selection. In: NIPS (2005)
13. Huang, J., Nie, F., Ding, C.: Robust manifold nonnegative matrix factorization. ACM Trans. Knowl. Discov. Data **8**, 11:1–11:21 (2014)
14. Li, J., et al.: Feature selection: a data perspective. ACM Comput. Surv. **50**, 94:1–94:45 (2017)
15. Li, J., Hu, X., Wu, L., Liu, H.: Robust unsupervised feature selection on networked data. In: SDM (2016)
16. Li, Z., Yang, Y., Zhou, X., Lu, H.: Unsupervised feature selection using nonnegative spectral analysis. In: AAAI (2012)
17. Lin, C.: Projected gradient methods for nonnegative matrix factorization. Neural Comput. **19**, 2756–2779 (2007)
18. Liu, H., Shao, M., Fu, Y.: Consensus guided unsupervised feature selection. In: AAAI (2016)
19. Liu, J., Ji, S., Ye, J.: Multi-task feature learning via efficient $\ell_{2,1}$-norm minimization. In: UAI (2009)
20. Liu, J., Ye, J.: Moreau-Yosida regularization for grouped tree structure learning. In: NIPS (2010)
21. Liu, Y., Wang, J., Ye, J.: An efficient algorithm for weak hierarchical lasso. ACM Trans. Knowl. Discov. Data **10**, 32:1–32:24 (2014)
22. Nie, F., Huang, H., Cai, X., Ding, C.H.: Efficient and robust feature selection via joint $ell_{2,1}$-norms minimization. In: NIPS (2010)

23. Pan, W., Yang, Q.: Transfer learning in heterogeneous collaborative filtering domains. Artif. Intell. **197**, 39–55 (2013)
24. Qian, M., Zhai, C.: Robust unsupervised feature selection. In: IJCAI (2013)
25. Tang, J.: Feature selection with linked data in social media. In: SDM (2012)
26. Tang, J., Li, H.: Unsupervised feature selection for linked social media data. In: KDD (2012)
27. Trigeorgis, G., Bousmalis, K., Zafeiriou, S.P., Schuller, B.W.: A deep semi-NMF model for learning hidden representations. In: ICML (2014)
28. Wang, S., Liu, H.: Embedded unsupervised feature selection. In: AAAI (2015)
29. Wang, S., Tang, J., Wang, Y., Liu, H.: Exploring implicit hierarchical structures for recommender systems. In: IJCAI (2015)
30. Wang, S., Wang, Y., Tang, J., Aggarwal, C., Ranganath, S., Liu, H.: Exploiting hierarchical structures for unsupervised feature selection. In: SDM (2017)
31. Wang, X., Tang, L., Gao, H., Liu, H.: Discovering overlapping groups in social media. In: ICDM (2010)
32. Wang, X., Tang, L., Liu, H., Wang, L.: Learning with multi-resolution overlapping communities. Knowl. Inf. Syst. **36**, 517–535 (2013). https://doi.org/10.1007/s10115-012-0555-0
33. Yang, Y., Shen, H., Ma, Z., Huang, Z., Zhou, X.: $\ell_{2,1}$-norm regularized discriminative feature selection for unsupervised learning. In: IJCAI (2011)
34. Ye, F., Chen, C., Zheng, Z.: Deep autoencoder-like nonnegative matrix factorization for community detection. In: CIKM (2018)
35. Yu, L., Liu, H.: Feature selection for high-dimensional data: a fast correlation-based filter solution. In: ICML (2003)
36. Zhao, Z., Liu, H.: Spectral feature selection for supervised and unsupervised learning. In: ICML (2007)