Proceedings of Machine Learning Research 80:1–16, 2018 ACML 2018

Clustering Uncertain Graphs with Node Attributes

Yafang Li YAFANGLI@BJUT.EDU.CN
Beijing University of Technology, Beijing, China

Xiangnan Kong XKONG@WPI.EDU
Worcester Polytechnic Institute, MA, USA

Caiyan Jia* CYJIA@BJTU.EDU.CN
Beijing Jiaotong University, Beijing, China

Jianqiang Li LIJIANQIANG@BJUT.EDU.CN
Beijing University of Technology, Beijing, China

Editors: Jun Zhu and Ichiro Takeuchi

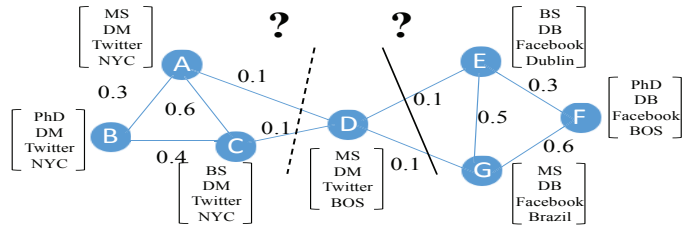
Abstract

Graph clustering has attracted much attention in recent years, which has wide applications in social and biological networks. Recent approaches on graph clustering mainly focus on either certain graphs with node attributes or uncertain graphs without node attributes. However, many real-world graphs have both uncertainty on the edges and attributes on the nodes. We refer to such networks as *attributed uncertain graphs*. Different from conventional graphs, attributed uncertain graphs post two major challenges for graph clustering: 1) uncertainty on the edges, which makes it difficult to extract reliable clusters; 2) high dimensional attributes on the nodes, which contain irrelevant and noisy information. In this paper, we study the problem of node clustering on attributed uncertain graphs, where we exploit both the uncertain edges and a set of important attributes for graph clustering. The uncertain edges can help identify the set of relevant attributes in the nodes, which are called focus attributes. While the focus attributes can help reduce the uncertainty in edges for graph clustering. We propose two novel approaches: AUG-I based upon integrated attribute induced graphs and AUG-U based upon the unified partition over possible worlds of a uncertain graph. Extensive empirical studies on real-world datasets demonstrate the effectiveness of our approaches for clustering tasks on attributed uncertain graphs.

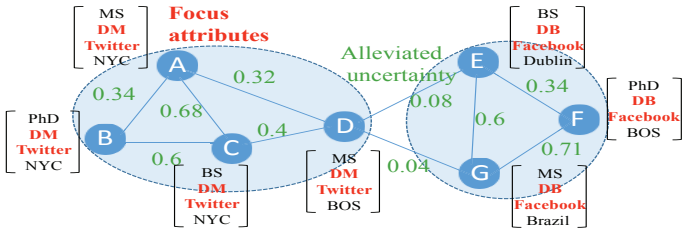
Keywords: Uncertain graphs, Node attributes, Graph clustering

1. Introduction

Many social and biological systems usually involve interconnected components. Such systems are often modeled as graphs, where nodes represent individuals, edges represent relationships between them, like friendship, kinship, interaction (Guimera and Amaral, 2005; Aggarwal and Reddy, 2013), *etc.* With the rapid development of Internet and mobile technology, such graph data are becoming easier to get access to. However, due to errors in data collection, privacy concerns, or preprocessing, such graph data are usually cluttered with inherent uncertainty on the edges. Every edge in these uncertain graphs is labeled with a probability of existence. For example, in protein-protein interaction networks, edges are usually experimentally inferred, which indicate the connection probability among proteins; in social networks, various complex relationships among users, such as influence and trust,



(*a*) An attributed uncertain graph



(*b*) Clusters extracted by our method

Figure 1: (a) is an example of attributed uncertain graph. Partitions by dashed line and solid line result in the same partition probability. (b) shows the clusters detected by our methods, which consider both alleviated uncertainty on the edges and focus attributes on the nodes.

are not directly observable, but are estimated indirectly from user activities. Such network data with uncertain edges are denoted as *uncertain graphs* or *probabilistic graphs*. Clustering uncertain graphs has recently gained popularity with widespread applications (Liu et al., 2012; Langohr and Toivonen, 2012; Kollios et al., 2013; Halim et al., 2015; Boonma and Natwichai, 2015; Ceccarello et al., 2017), such as identifying functional modules in biological networks, grouping authors sharing similar research interest in co-author networks.

Conventional approaches to uncertain graph clustering problems usually focus on only utilizing the uncertainty in graph structure. However, many real-world uncertain graphs have abundant attributes associated with the nodes, in addition to their uncertain connectivity information. For example, a protein-protein interaction network not only has interaction relations but also has the gene expressions associated with the proteins. Social networks contain uncertain edges as well as personal attribute information, including age, gender, interest, *etc.* We refer to these uncertain graphs with attributes on nodes as *attributed uncertain graphs*. To cluster these graphs, only considering the structural information is inadequate to accurately determine the community structure, due to the uncertainty on the edges. Hence, the abundant node attributes can be leveraged to reduce the edge uncertainty and identify desirable clusters. However, the attributes of nodes are usually high dimensional which contain irrelevant and noisy information. If we use all attributes for clustering, irrelevant attributes can significantly harm the clustering performance.

In order to effectively cluster the nodes in attributed uncertain graphs, we need to exploit both the uncertainty on the edges and the relevant attributes on the nodes. We give an example of the relationship network in Fig. 1 (a). In this graph, seven users (marked as A to G) are linked with a probability representing the degree of collaboration with each other. Each user is also associated with a list of features denoting *research interest*, *work*, and *location*, *etc.* If we ignore the attribute information in this uncertain graph, with the

possible world semantics (Kollios et al., 2013; Liu et al., 2012), the probability to partition the graph with the dashed line is $(1-0.1)^2 = 0.81$. To separate with the solid line is also $(1-0.1)^2 = 0.81$. In this case, it is hard to decide which is the desirable partition. If we take all of the attributes into account, E and G might be grouped separately as a result of the major difference in affiliation.

To address the above challenges in clustering attributed uncertain graphs, in this study, we propose two clustering methods, AUG-I based upon integrated attribute induced graphs, and AUG-U based on the unified partition over all possible worlds of a uncertain graph. The main idea is to use a subset of relevant attributes to alleviate the edge uncertainty so as to find reliable clusters, where nodes are unlikely to be disconnected, besides, nodes in the same cluster are of high semantic homogeneity. To be specific, the literature on uncertain graphs assumes the existence of the edges in the graph are independent from one another. Thus, in this study, to deal with the edge uncertainty, we adopt the well-known *possible-world semantics* model (Liu et al., 2012; Kollios et al., 2013), which generates possible deterministic attributed graphs by independently sampling each edge with its appearance probability in an attributed uncertain graph. Each possible deterministic graph is assumed to be a *view* of the true graph structure, which captures a particular aspect of node connections. Also, nodes in each deterministic graph are embedded with abundant attributes describing personal features. We assume that nodes in each deterministic attributed graph are connected for the reason that nodes share high similarity in subset of relevant attributes, called focus attributes. Start with this, these focus attributes are inferred to enrich the connections between nodes to alleviate edge uncertainty. With the set of attribute induced graphs, to extract clusters, we propose AUG-I method by integrating all attribute refined graphs. We also present AUG-U to partition nodes by a single unified cut over all attribute induced graphs. An example by considering node attributes is shown in Fig. 1 (b), where every edge is refined and assigned with a new weight. Specifically, nodes with common *research interest* and *work* are linked with a large weight, like the edges between A and D, C and D. Otherwise, the edge has a smaller weight, like D and E, D and G. After refinement, nodes circled are grouped into the same clusters shown in Fig. 1 (b).

We summarize our contributions below:

1) To the best of our knowledge, this is the first work to define *attributed uncertain graphs* and exploit the problem of clustering uncertain graphs with node attributes.

2) We present two novel methods, AUG-I and AUG-U, to address the challenges in attributed uncertain graph clustering, i.e., uncertainty of edges and high dimensionality of attributes.

3) We conduct extensive studies to show the effectiveness of our proposed approaches.

2. Related work

Our work is related to both uncertain graph clustering and attributed graph clustering. We briefly discuss both of them.

Clustering uncertain graphs is to find clusters in graphs with probabilistic edges. Currently, some efforts have been made in uncertain graphs clustering (Liu et al., 2012; Langohr and Toivonen, 2012; Kollios et al., 2013; Halim et al., 2015; Boonma and Natwichai, 2015; Ceccarello et al., 2017). A representative approach, coded kmeans, is presented by Liu *et*

al. (Liu et al., 2012). They use coding methods from the information theory to represent the structures in clustering process. Authors in (Langohr and Toivonen, 2012) defined a probabilistic similarity measure for nodes and used it to find clusters among groups of nodes by conventional clustering methods. Kollios *et al.* extend the edit-distance based definition of graph clustering to uncertain graphs (Kollios et al., 2013). Halim *et al.* cluster uncertain graphs based on multi-population evolutionary algorithm (Halim et al., 2015). Ceccarello *et al.* partition nodes by maximizing the minimum/average connection probability of any node to its cluster center (Ceccarello et al., 2017). However, all of these methods are limited to uncertain graphs without attributes.

Attributed graph clustering has been widely investigated by many researcher in recent years (Zhou et al., 2009; Yang et al., 2009, 2013; Rafailidis, 2016; Jia et al., 2017). They aim to partition the given graph into groups, in which nodes are cohesive connected and attribute-wise homogeneous. However, these methods enforce attribute homogeneity in all attributes. Though some methods loosen this constraint by subspace clustering and extract dense subgraphs with semantic similarity in a subset of attributes (Moser et al., 2009; Günnemann et al., 2010; Perozzi et al., 2014), all of these methods are designed for conventional deterministic graphs, and they can not directly applied for attributed graphs with uncertain structures. If we treat the attributed uncertain graph as attributed weighted graphs by casting edge probability into edge weight. It will be problematic that it fails to reflect the connectivity of uncertain graphs correctly.

3. Problem formulation

In this study, we introduce the novel problem of focused clustering in attributed uncertain graphs. We first give the basic definitions of an attributed uncertain graph and induced deterministic attributed graphs.

Definition 1 (Attributed uncertain graph) *An attributed uncertain graph is represented as $\mathcal{G} = (V, E, F, P)$, where V corresponds to the set of n nodes in $\mathcal{G}$; $E \in V \times V$ denotes the undirected edges between nodes; F is a set of n attribute vectors, which indicate the d attributes associated with each node; P maps every pair of nodes to a real number in the interval $[0, 1]$; p_{uv} represent the probability that the edge $(u, v) \in E$ exists.*

Definition 2 (Deterministic attributed graph) *A deterministic attributed graph $G = (V_S, E_S, F_S)$ is a particular case of an attributed uncertain graph, where edges show a binary relationship between nodes. Each deterministic attributed graph G is achieved by sampling each edge $(u, v) \in E$ in $\mathcal{G}$ according to its probability p_{uv}, denoted as $G \sqsubseteq \mathcal{G}$.*

For an uncertain network, *reliability* (Colbourn, 1987) is widely used to capture the probability that a set of vertices are connected. It generalizes the connectivity to the probabilistic scenario. The reliability for vertex set $V_s \subseteq V$ of an uncertain graph $\mathcal{G}$ is defined as

$$R(V_s) = \sum_{G_i \sqsubseteq \mathcal{G}} Pr(G_i)\mathcal{I}(V_s, G_i),$$

where G_i is a deterministic graph generated from $\mathcal{G}$, $Pr(G_i)$ is the sampling probability. $\mathcal{I}(V_s, G_i)$ is 1, if V_s is contained in a connected component in G_i, and 0 otherwise. The reli-

ability value of an uncertain graph lies in the interval $(0, 1)$, which quantifies the probability of the vertices remaining connected in an uncertain graph.

Given an attributed uncertain graph $\mathcal{G}$, our purpose is to extract k clusters that are: 1) high reliability and structurally dense; (2) homogenous on the relevant attributes. We assume that nodes connected in the deterministic attributed graph G_i are similar to each other and share high semantic similarity in certain axes from high feature dimensionality, as nodes usually have many noisy and irrelevant attributes. From this point, we aim to infer the implicit attribute weights $\boldsymbol{\beta}_i$ that qualify the similarity of nodes in a subset of feature dimensions. Thus, $\boldsymbol{\beta}_i$ is expected to be a sparse vector with large weights for only a few attributes.

Having learnt the attribute weights $\boldsymbol{\beta}_i$ in each deterministic attributed graph, our first goal is to build a weighted graph, denoted as *attribute induced graph*. It is a key issue of utilizing the associated relevant attributes to refine structural connections. In this way, edge uncertainty is alleviated by consistence of attribute with large weights. With attributes refined graphs, we present two methods, AUG-I and AUG-U, to partition the graphs.

4. Proposed methods

In this section we first introduce our sampling deterministic attributed graphs from $\mathcal{G}$. Then we present how to alleviate edge uncertainty between nodes with attribute weights. Next we introduce our AUG-I and AUG-U methods to partition the graphs.

4.1. Sample deterministic attributed graphs

We adopt the possible-world semantic model to inject an uncertain attributed graph $\mathcal{G} = (V, E, F, P)$ to deterministic attributed graphs $G = (V_S, E_S, F_S)$ by sampling each edge (u, v) independently according to its existence probability p_{uv}. The larger the probability p_{uv} is, the higher possibility (u, v) is present in the deterministic attributed graph G. The probability of sampling a deterministic attributed graph is:

$$Pr(G) = \prod_{\{u,v\} \in E_S} p_{uv} \prod_{\{u,v\} \in E \setminus E_S} (1 - p_{uv}).$$

The deterministic attributed graphs generated from $\mathcal{G}$ have binary values on edges, which imply the possible definite connections of nodes. If nodes are connected, it is 1, otherwise, 0. Each deterministic graph G is a particular case of $\mathcal{G}$ according to the probability distribution $Pr(G)$. Hence, each sampled deterministic attributed graph can be regarded as a *view* of $\mathcal{G}$ and observed with the probability $Pr(G)$.

4.2. Attribute induced graphs

For $G \sqsubseteq \mathcal{G}$, the sampled deterministic graph is assumed to reflect the true connections among nodes to some degree. However, there still exists some noise in the linkage, as a result of low probability edge will be selected in the deterministic scenario. Hence, we expect to utilize the rich attributes of nodes to refine the connections and construct a new weighted graph.

Based on the edges between nodes, our first goal is to infer the relevance weights of node attributes that make the nodes similar and connect to each other in this deterministic graph. To achieve this, we capture this weighted similarity by the (inverse) Mahalanobis distance, which measures the distance between nodes u and v with their attribute feature vector $\boldsymbol{f_u}$ and $\boldsymbol{f_v}$ by $(\boldsymbol{f_u} - \boldsymbol{f_v})^\top \boldsymbol{W}(\boldsymbol{f_u} - \boldsymbol{f_v})$. If we restrict $\boldsymbol{W}$ to be diagonal, this corresponds to learning a weight in each attribute dimension. Specially, if we set $\boldsymbol{W} = I$, the distance is degenerated into Euclidean distance.

Given the deterministic graph, we adopt the distance metric learning (Xing et al., 2003) to infer $\boldsymbol{W}$ such that two connected nodes have small distance to each other. The learning problem can be formulated as follows:

$$\min_{\boldsymbol{W}} \sum_{(u,v)\in\mathcal{S}} ||\boldsymbol{f_u} - \boldsymbol{f_v}||^2_{\boldsymbol{W}} - \gamma \log \Bigg(\sum_{(u,v)\in D} ||\boldsymbol{f_u} - \boldsymbol{f_v}||_{\boldsymbol{W}} \Bigg), \tag{1}$$

which is a convex optimization problem and it enables to be solved by efficient, local-minima-free algorithms, especially for a diagonal solution.

$\bar{G}$=AttInduce(G)

Input:

G_i : Deterministic attributed graph $G_i = \{V_S, E_S, F_S\}$;
α : A threshold that filter weighted edges;

Process:

// Create must-link and cannot-link node pairs;
1 $S = \emptyset$, $D = \emptyset$;
2 $S \leftarrow E_S$;
3 **repeat**
4 Random sample unconnected node pair $\{u, v\} \setminus E_S$, $D \leftarrow D \cup (u, v)$;
5 **until** $|D| = |S|$;
6 Solve function in Eq. 1 for diagonal $\boldsymbol{W}$;
7 $\boldsymbol{\beta} \leftarrow diag(\boldsymbol{W})$;
8 **for** edge $(u, v) \in E_S$ **do**
9 $E_S[w(u, v)] \leftarrow \frac{1}{(1+\sqrt{(f_u - f_v)^\top diag(\boldsymbol{\beta})(f_u - f_v)})}$
10 **end for**
11 $\bar{E}_S \leftarrow \emptyset$;
12 **for all** edge $(u, v) \in E_S$ **do**
13 $Seq \leftarrow sort(E_S[w(u, v)])$
14 **end**
15 **if** $E_S[w(u, v)] > \alpha|E_S|$ **do**
16 $\bar{E}_S \leftarrow \bar{E}_S \cup (u, v)$;
17 **end if**
18 $\bar{G} \leftarrow (V_S, \bar{E}_S)$;

Output:

$\bar{G}$: the attribute induced weighted graph.

Figure 2: Construct attribute induced graph.

The details of inferring attribute weights is given in Fig. 2 (Lines 1-7), where S and D are respectively the must-link and cannot-link pairs of nodes. In our setting, all the connected node pairs in the deterministic graph constitute S. This is a suitable assumption, nodes linked with a larger probability in $\mathcal{G}$ have a higher chance to be selected as a pair of connected nodes in each possible graph. Thus, the connectivity is a reflection of similarity

$\mathcal{C}$ = AUG-I($\mathcal{G}$, N, k)

Input:

$\mathcal{G}$: Attributed uncertain graph $\mathcal{G} = (V, E, F, P)$
N : Number of sampled deterministic graphs
k : Number of clusters

Process:

$E_e \leftarrow \emptyset$;
for $1 \leq i \leq N$ **do**
  $Pr(G_i) = \prod_{\{u,v\} \in E_S} p_{uv} \prod_{\{u,v\} \in E \setminus E_S} (1 - p_{uv})$;
  $\bar{G}_i$=*AttInduce*(G_i);
  for $(u, v) \in \bar{E}_S$ **do**
    if $(u, v) \notin E_e$ **then**
    $E_e \leftarrow E_e \cup (u, v)$;
    end if
    $E_e[w(u, v)] \leftarrow E_e[w(u, v)] + \frac{Pr(G_i)\bar{E}_S[w(u,v)]}{\sum_{i=1}^{N} Pr(G_i)}$;
  end for
end for
$\mathcal{G}_e \leftarrow (E_e, V)$;
$Aij \leftarrow \begin{cases} E_e[w(u, v)], (u, v) \in E_e \\ 0, otherwise. \end{cases}$
$D_{ii} \leftarrow \sum_{j=1}^{|V|} A_{ij}$;
$\boldsymbol{L} = \boldsymbol{I} - \boldsymbol{D}^{-1/2}\boldsymbol{A}\boldsymbol{D}^{-1/2}$;
$\{C_1, C_2, \ldots, C_k\} \leftarrow$ *Spectral*($\boldsymbol{L}, k$);
return $\mathcal{C}$;

Output:

$\mathcal{C}$: Set of k clusters

Figure 3: The AUG-I method

between nodes, if there exists no edge between two nodes, it means they are dissimilar with each other. With this observation, we create D by randomly drawing pairs of nodes that do not connect with each other. Meanwhile, to alleviate the affect of unbalance size of selected node pairs, we keep $|D| = |S|$.

Having inferred the attribute weights, we reweigh the edges in G with the weighted similarity of end nodes (Lines 8-10 of Fig. 2), which is induced by attributes of nodes with large weights. However, as discussed above, noisy edges may be selected in the deterministic graph. Hence, to further refine the attribute induced graph, we filter edges under a weight threshold and construct a new graph $\bar{G}$ (Lines 15-18 of Fig. 2).

4.3. AUG-I

Given a set of attribute induced graphs, to partition the original uncertain graph, an intuitive way is to integrate all graphs into a combination one. Following this idea, we propose our first method, called AUG-I. We note that each deterministic attributed graph G_i can be taken as an instance of the original attributed uncertain graph $\mathcal{G}$ according to its sampling probability $Pr(G_i)$. Accordingly, the attribute induced graph $\bar{G}_i$ from G_i is proportional to the probability distribution $Pr(G_i)$. By refining edges with attributes of large weights, each induced graph $\bar{G}_i$ is assumed to be a particular *view* of the uncertain graph structure, which captures a certain aspect of the true connections between nodes. To integrate all possible aspects of true connections and get an overview observation, we combine all attribute

induced graphs (*views*) to build a new weighted graph. This is done by an average weighted combination according to its probability distribution as described in Fig. 3 (Lines 5-10). After integration, we identify clusters in this weighted graph, denoted as $\mathcal{G}_e$, by normalized spectral clustering shown in Fig. 3 Lines 13-16.

4.4. AUG-U

We also propose a unified partition method, called AUG-U, which aims to find a single cut on all induced graphs, instead of a cut on an integrated graph. Given the set of attribute induced graphs $\{\bar{G}_1, \bar{G}_2, \ldots, \bar{G}_N\}$, correspondingly, $\{\bar{\boldsymbol{A}}_1, \bar{\boldsymbol{A}}_2, \ldots, \bar{\boldsymbol{A}}_N\}$ is the collection of adjacency matrices. Assuming each individual induced graph has an optimal cut partitioning the graph with minimum cost, the single cut should satisfy the following criterions:

- It has low cut costs on on all attribute induced graphs.
- It is similar to each induced graph's individual cut.

The first goal requires the costs of the final single cut over all induced graphs is small. To achieve this, we average the weighted cut costs on all graphs according to the sampling probability of each individual graph. As each attribute induced graph $\bar{G}_i$ is associated with a sample probability $Pr(G_i)$, which reflects the possibility nodes are connected, further, it influences the cut costs with the single unified cut. We note that the cost of a cut v_i on a graph $\bar{G}_i$ is defined typically in clustering $C(\boldsymbol{v}_i, \boldsymbol{A}_i) = \boldsymbol{v}_i^\top \boldsymbol{L}_i \boldsymbol{v}_i$, where $\boldsymbol{L}_i = \boldsymbol{I} - \boldsymbol{D}_i^{-1/2} \boldsymbol{A}_i \boldsymbol{D}_i^{-1/2}$ is the normalized Laplacian matrix of each induced graph, where $\boldsymbol{A}_i = Pr(G_i)\bar{\boldsymbol{A}}_i$, correspondingly, $\boldsymbol{D}_i$ is the degree matrix of $\boldsymbol{A}_i$. Hence, the cut of each graph $\bar{G}_i$ can be computed typically as the eigenvector $\boldsymbol{v}_i$ of the Laplacian matrix $\boldsymbol{L}_i$ corresponding to the second small eigenvalue. We assume the single cut on all graphs is denoted as $\boldsymbol{u}$. Therefore, the total cost of the single cut on all induced graphs is the summation of all costs $\mathcal{L}_c = \sum_{i=1}^{N} \boldsymbol{u}^\top \boldsymbol{L}_i \boldsymbol{u}$.

The second criterion is to make sure the single cut is similar to each individual cut on all induced graphs. As each individual cut can be viewed as a eigenvector $\boldsymbol{v}_i \in \mathbb{R}^n$. Essentially, to evaluate the similarity between the single cut and the optimal cuts on all induced graphs, a natural way is to measure the cosine similarity with separate cut on each individual graph $\frac{\boldsymbol{u}^\top \boldsymbol{v}_i}{||\boldsymbol{u}||_2^2 ||\boldsymbol{v}_i||_2^2}$. As each vector is normalized into an unit vector, besides, maximizing this is equivalent to maximizing the square of the value. Hence, the overall similarity over all graphs is $\mathcal{L}_s = \sum_{i=1}^{N} (\boldsymbol{u}^\top \boldsymbol{v}_i)^2$.

We combine the aforementioned motivations into the following problem:

$$\begin{aligned} \min_{\boldsymbol{u}} \quad & \sum_{i=1}^{N} \boldsymbol{u}^\top \boldsymbol{L}_i \boldsymbol{u} - \delta \sum_{i=1}^{N} (\boldsymbol{u}^\top \boldsymbol{v}_i)^2 \\ s.t. \quad & \boldsymbol{u}^\top \boldsymbol{u} = 1 \end{aligned}$$

where δ is a weighting parameter, which controls the trade-off between these two objective terms. Note that above problem is equivalent to minimization of the following problem:

$$\min_{\boldsymbol{u}} \quad \boldsymbol{u}^\top \big[\sum_{i=1}^{N} (\boldsymbol{L}_i - \delta \boldsymbol{v}_i \boldsymbol{v}_i^\top) \big] \boldsymbol{u}$$

To solve above constrained optimization problem, Karush-Kuhn-Tucker (KKT) conditions can be utilized, then, the Lagrangian is defined as:

$$\mathcal{L} = \boldsymbol{u}^\top \big[\sum_{i=1}^{N} (\boldsymbol{L}_i - \delta \boldsymbol{v}_i \boldsymbol{v}_i^\top) \big] \boldsymbol{u} - \theta(\boldsymbol{u}^\top \boldsymbol{u} - 1)$$

Taking the stationarity condition, we get:

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{u}} = (\sum_{i=1}^{N} (\boldsymbol{L}_i - \delta \boldsymbol{v}_i \boldsymbol{v}_i^\top)) \boldsymbol{u} - \theta \boldsymbol{u} = 0$$

From primal feasibility condition, we have:

$$\frac{\partial \mathcal{L}}{\partial \theta} = \boldsymbol{u}^\top \boldsymbol{u} - 1 = 0$$

Taking $\boldsymbol{M} = \sum_{i=1}^{N} (\boldsymbol{L}_i - \delta \boldsymbol{v}_i \boldsymbol{v}_i^\top)$, we note that the candidate solutions are the eigenvectors of $\boldsymbol{M}$, the eigenvector corresponding to the smallest eigenvalue is the optimal solution as shown in Fig. 4.

$\mathcal{C}$ = AUG-U($\mathcal{G}$, N, k)

Input:

$\mathcal{G}$: Attributed uncertain graph $\mathcal{G} = (V, E, F, P)$
N : Number of sampled deterministic graphs
k : Number of clusters

Process:

$E_e \leftarrow \emptyset$;
for $1 \le i \le N$ **do**
$Pr(G_i) = \prod_{\{u,v\} \in E_S} p_{uv} \prod_{\{u,v\} \in E \setminus E_S} (1 - p_{uv})$;
$\bar{G}_i$=*AttInduce*(G_i);
$Aij \leftarrow \begin{cases} Pr(G_i) E_e[w(u,v)], (u,v) \in E_e \\ 0, otherwise. \end{cases}$
$D_{ii} \leftarrow \sum_{j=1}^{|V|} A_{ij}$;
$\boldsymbol{L}_i \leftarrow \boldsymbol{I} - \boldsymbol{D}_i^{-1/2} \boldsymbol{A}_i \boldsymbol{D}_i^{-1/2}$;
end for
$\boldsymbol{M} \leftarrow \sum_{i=1}^{N} (\boldsymbol{L}_i - \delta \boldsymbol{v}_i \boldsymbol{v}_i^\top)$; // $\boldsymbol{v}_i$ is the eigenvector of an individual cut
$\{C_1, C_2, \ldots, C_k\} \leftarrow$ *Spectral*($\boldsymbol{M}, k$);
return $\mathcal{C}$;

Output:

$\mathcal{C}$: Set of k clusters

Figure 4: The AUG-U method

4.5. Complexity analysis

Given an attributed uncertain graph $\mathcal{G}$ with n nodes and m edges, we generate N possible deterministic attributed graphs. For each sampled graph, we infer the weights by optimizing the objective function in Eq. 1, we aim for a diagonal solution, local-optima-free gradient descent techniques will take $O(\frac{d}{\epsilon^2})$ for an ϵ-approximate answer (Boyd and Vandenberghe,

2004). Having get the weight vector $\boldsymbol{\beta}$, to assign each edge a new weight with complexity $O(md)$. We assume only a small set of attributes are highly relevant to make nodes similar to each other. Thus, $\boldsymbol{\beta}$ is sparse with only a few non-zero elements for these attributes. In this case, the multiplicative factor becomes effectively constant yielding a complexity of $O(m)$. Next, we keep the top $\alpha|E_S|$ edges with large weights to construct a new graph. To do so, we adopt minwise hasing to sort the edges incident to each node by weights. This can be done in linear time and require $O(m)$.

For AUG-I, to integrate all possible deterministic graphs, this takes $O(Nm)$ in the worst case, where N is the total number of deterministic graphs, as a portion of edges are filtered when constructing attribute induced graphs. For AUG-U and extracting clusters by spectral clustering algorithm, the most expensive step is the computation of the eigenvectors of Laplacian matrix. To speed up for large graph, we compute the first k' eigenvectors by a sampling technique to find an approximate solution with Nyström method, the complexity is $O(l^3) + O(nlk')$, where $l \ll n$ is the number of sampled points (Chen et al., 2011). With the obtained matrix built from eigenvectors, it requires $O(nk'^2)$ to partition with kmeans algorithm. When we use parallel kmeans, it takes $O(\frac{nk'^2}{p})$, where p is the number of local machines. Besides, we notice that the operation on each sampled deterministic attributed graph is independent, which allows for parallel processing for speed up.

5. Experiment

To examine the performance in attributed uncertain graph clustering, we conducted extensive experiments on synthetic and real-world datasets. In this section, we introduced the datasets we used and the experiments we performed respectively, then we presented the experimental results as well as the analysis, we also discussed the parameter sensitivity.

5.1. Data Collection

We used two collections of datasets: synthetic networks and real-world networks.

Synthetic datasets: A synthetic network involves two information: edge uncertainty and node attributes. To generate the topological structure, we use LFR benchmark networks Lancichinetti et al. (2008), which possess some basic statistical properties found in many real-world networks, such as power law distribution of the degree and community size. Besides, it specifies a mixing parameter μ such that every node connects a fraction of μ nodes outside its cluster. The smaller μ is, the clearer community structure is. In this study, we generate three LFR benchmarks setting $\mu = [0.6, 0.7, 0.8]$, other parameters are: network size is set to 50, average degree $\langle d_k \rangle = 5$, maximum degree $k_{max} = 20$, $\tau_1 = 2$, $\tau_2 = 1$, cluster size is in a range of $[10, 20]$.

We also associate attributes for each node in the network. We define that each node is described as one binary vector indicating the presence (the value is 1) or absence (the value is 0) of the corresponding attribute from a collection of f_a attributes. For nodes in the same cluster, they are characterized by the same f_c attributes, which describe the cluster profiles. To blur the feature description, we add a percentage (ρ) of randomly chosen irrelative attributes to each node. Specifically, in this study, the parameters are set as: $f_a = 100$, $f_c = 30$, and the noise rate is $\rho = 0.4$.

Real-world datasets: To evaluate performances for attributed uncertain graph clustering, we compared our methods against baselines on five real-world graphs. One is Cora dataset, a citation graph containing 2708 research papers from seven subfields of machine learning. There are overall 5429 links indicating the citation relationships among these papers. In this graph, each node is characterized by a 0/1-valued word vector indicating the absence/presence of the corresponding word from the dictionary, which consists of 1433 unique words. Another set of graphs contain WWW-pages collected from the computer science departments of four universities: Cornell, Texas, Washington, and Wisconsin (Sen et al., 2008). Each page network is manually classified into five classes: course, faculty, student, project, and staff. These four universities have 195, 187, 230, and 265 nodes and 304, 328, 446, and 530 edges, respectively. The edge probability is randomly generated from a uniform distribution in the interval $[0, 1]$ as in (Liu et al., 2012).

Table 1: Summary of experimental datasets in studied. "Avg(p)" denotes the average edge probability of each uncertain graph.

Dataset	n	# Edge	d	Avg(p)
Cornell	195	304	1703	0.4964
Texas	187	328	1703	0.5161
Washington	230	446	1703	0.4992
Wisconsin	265	530	1626	0.4878
Cora	2708	5429	1433	0.5035

5.2. Evaluation Metrics

To test the performances of different methods in clustering attributed uncertain graphs, we adopt accuracy (ACC) (Strehl and Ghosh, 2003) and pairwise F-measure (PWF) (Yang et al., 2009) to evaluate the clustering accuracy of different methods. We also use the average clustering reliability (ACR) (Liu et al., 2012) to measure the reliability of each cluster obtained by different methods. ACR quantifies the probability of nodes remaining connected in each cluster. Supposing $\mathcal{C} = (C_1, C_2, \ldots, C_k)$ is the k clusters in an attributed uncertain graph with n nodes. ACR is defined as:

$$ACR(\mathcal{C}) = \frac{\sum_{i=1}^{k} |C_i| R(C_i)}{n}.$$

5.3. Comparing Methods

In order to examine the effectiveness of our approaches, we compare against the following methods:

• Coded-kmeans (Liu et al., 2012): A presentative method that exploits edge uncertain in graph clustering.

• Spectral clustering (Ng et al., 2002): In our proposed methods, we adopt Spectral clustering to partition the graph in the last step, so we compare with the Spectral clustering method. It transfers the uncertain graphs into deterministic graphs by taking the uncertainty of each edge as edge weight.

• FocusCO (Perozzi et al., 2014): An attributed graph clustering method that considers subset of attributes and ignores edge uncertainty.

• AUG-I (this paper): It considers both edges uncertainty and node attributes. It extracts clusters on an integrated weighted graphs by merging all attribute induced graphs.
• AUG-U (this paper): It exploits both information. It finds a unified single cut that is similar to each individual cut on each attribute induced graph.

5.4. Performance on synthetic graphs

Table 2 demonstrates the empirical results of different methods on three synthetic datasets. Based on the results, we can notice that our presented AUG-I method significantly outperforms baseline on most metrics with different settings of μ, especially when $\mu = 0.6$. The results of our another method AUG-U comes the next. This indicates the effectiveness of our methods to consider both edge uncertainty and node attributes. Besides, it proves that by integrating all particular views of the original uncertain graph in clustering task, compared with AUG-U, which finds a unified cut to balance the results on all possible worlds, AUG-I captures more embedding network and performs better. Of all these methods, spectral method demonstrates the worst results, which may be caused by simply transferring uncertainty into edge weights. Besides, coded kmeans and FocusCo have better results and show similar performances on these datasets, however, their results are worse than our methods. This may be due to exploiting only one of these two information, either edge uncertainty or node attributes. It indicates that only considering one information is not adequate to obtain desirable results.

Table 2: Comparison results of different methods on the synthetic dataset. "Avg." denotes the average rank of different methods with different measurements.

μ	methods	evaluation			Avg.
		ACC	PWF	ACR	
0.6	AUG-I	0.520 (1)	0.501 (1)	0.090 (2)	1.3
	AUG-U	0.440 (2)	0.500 (2)	0.097 (1)	1.6
	spectral	0.360 (4)	0.325 (4)	0.001 (4)	4
	coded kmeans	0.440 (2)	0.334 (3)	0.001 (4)	3
	FocusCO	0.410 (3)	0.311 (5)	0.008 (3)	3.6
0.7	AUG-I	0.480 (2)	0.523 (1)	0.082 (1)	1.3
	AUG-U	0.460 (3)	0.508 (2)	0.056 (2)	2.3
	spectral	0.500 (1)	0.501 (3)	0.011 (5)	3
	coded kmeans	0.460 (3)	0.333 (5)	0.012 (4)	4
	FocusCO	0.400 (4)	0.404 (4)	0.047 (3)	3.6
0.8	AUG-I	0.380 (1)	0.356 (1)	0.119 (1)	1
	AUG-U	0.340 (3)	0.345 (2)	0.107 (2)	2.3
	spectral	0.300 (5)	0.262 (4)	0.096 (3)	4
	coded kmeans	0.360 (2)	0.249 (5)	0.059 (4)	3.6
	FocusCO	0.312 (4)	0.308 (3)	0.054 (5)	4

5.5. Performance on real-world graphs

Empirical results of different methods on real datasets are given in Table 3. Based on the results, we have the similar conclusion with the synthetic datasets, namely, our proposed AUG-I and AUG-U methods generally outperform the other baselines on almost all datasets in terms of all metrics. This further proves the effective use of relevant attributes to refine

the uncertain structure such that edge uncertainty is alleviated. Besides, we have made following interesting observations:

• Coded kmeans finds many separated loosely-connected clusters with lower ACR values. This may be caused by basing on disconnected "fragments" (components) in each possible graphs. Another possible reason is that it ignores the abundant node attributes in helping to refine node connections.

• Different from coded kmeans, FocusCO is inclined to discover more connected clusters, resulting in a large ACR value, especially on Cornell dataset showing the best result. Nevertheless, it partitions the graph into many small cohesive components and has low clustering accuracy. This may largely due to overlooking the edge uncertainty and not capture the real topological structure.

• By considering both edge uncertainty and node attributes, the overall accuracy of our proposed AUG-I and AUG-U methods have improved. What is more, AUG-U tends to extract more connected clusters and AUG-I have better clustering accuracy. We can conclude that integrating all attributed graphs can better capture the real graph structure.

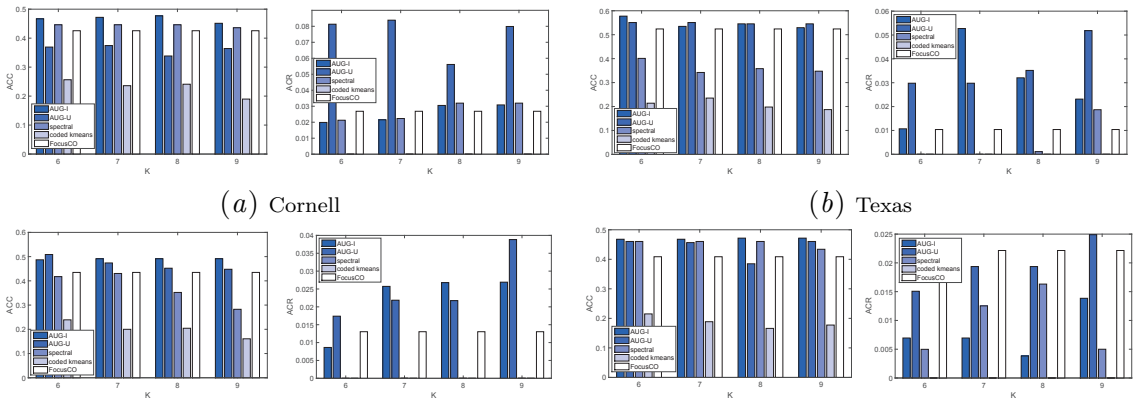
Table 3: Results of different methods on the real-world datasets. "Avg." denotes the average rank of methods on datasets.

criteria	methods	datasets					Avg.
		cornell	texas	washington	wisconsin	cora	
ACC ↑	AUG-I	0.482 (1)	0.593 (1)	0.509 (1)	0.468 (1)	0.313 (1)	1
	AUG-U	0.426 (3)	0.551 (2)	0.474 (2)	0.449 (3)	0.290 (2)	2.4
	spectral	0.441 (2)	0.395 (4)	0.413 (3)	0.460 (2)	0.289 (3)	2.8
	coded kmeans	0.267 (5)	0.247 (5)	0.257 (5)	0.253 (5)	0.164 (4)	4.8
	FocusCO	0.353 (4)	0.433 (3)	0.391 (4)	0.316 (4)	0.289 (3)	3.6
PWF ↑	AUG-I	0.442 (1)	0.543 (1)	0.470 (2)	0.457 (3)	0.292 (1)	1.6
	AUG-U	0.420 (3)	0.540 (2)	0.484 (1)	0.462 (2)	0.290 (2)	2
	spectral	0.433 (2)	0.357 (4)	0.395 (3)	0.474 (1)	0.289 (3)	2.8
	coded kmeans	0.230 (5)	0.253 (5)	0.246 (5)	0.245 (5)	0.159 (5)	5
	FocusCO	0.349 (4)	0.413 (3)	0.355 (4)	0.348 (4)	0.288 (4)	3.8
ACR ↑	AUG-I	0.019 (3)	0.009 (3)	0.017 (1)	0.008 (3)	0.001 (1)	2.2
	AUG-U	0.021 (2)	0.032 (1)	0.017 (1)	0.016 (1)	0.000 (2)	1.4
	spectral	0.011 (4)	0.000 (4)	0.000 (3)	0.000 (4)	0.000 (2)	3.4
	coded kmeans	0.000 (5)	0.000 (5)	0.000 (4)	0.000 (4)	0.000 (2)	4
	FocusCO	0.026 (1)	0.010 (2)	0.013 (2)	0.012 (2)	0.000 (2)	1.8

To further prove the effectiveness of our proposed methods, we compared against baseline methods with different settings of the number of clusters K shown in Figs. 5(a) - 5(d). From these figures, we can draw similar conclusion as the results in Table 3. We further confirm that our presented AUG-I and AUG-U methods are effective in clustering attributed uncertain graphs. If only taking one source of information into account, the edge uncertainty or the node attributes, it is not enough to achieve optimal results.

5.6. Parameter studies

There exists two essential parameters in our proposed methods, sample size N and filter threshold α. Fig. 6 shows the sensitivity of AUG-I upon different settings of N. Different from coded kmeans, requiring the sample size at least to be 5600 suggested in that work,

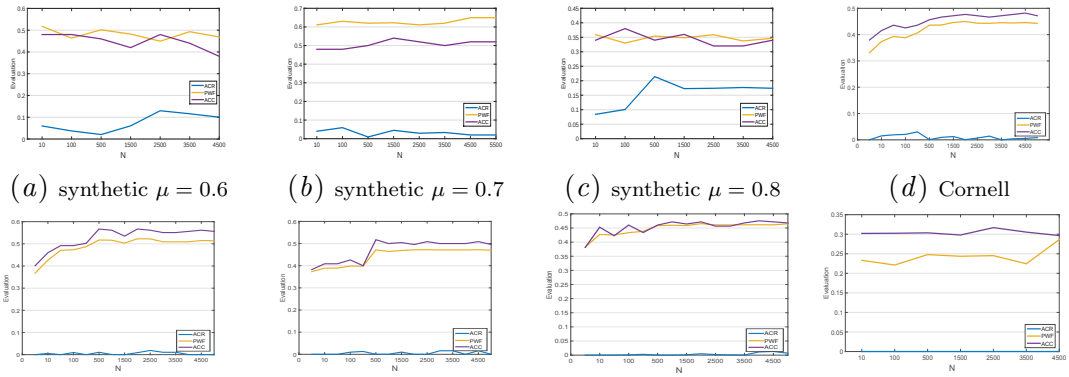


(a) Cornell (b) Texas

(c) Washington (d) Wisconsin

Figure 5: Results of different number of clusters k.

generally, AUG-I is not very sensitive to the value of N once chosen a relatively large N. We have a similar observation for AUG-U and we do not report it here for length limit. According to the figures, we find that a small sample size usually leads to a non-optimal partition result. This is due to the sampled deterministic graphs are inadequate to capture the true connections between nodes. Intuitively, the larger sample size N, the more information can be captured by the sampled deterministic graphs. As shown in these figures, the performances on all datasets are gradually improved with the increase of N. When N achieves 1500, the clustering quality on these four data sets gets more stable and optimal, as to Texas and Washington datasets, optimal results can be obtained when $N \approx 500$. The results show the robustness of AUG-I, where its performance remains quite stable once the sample size reaches a certain not too large value.



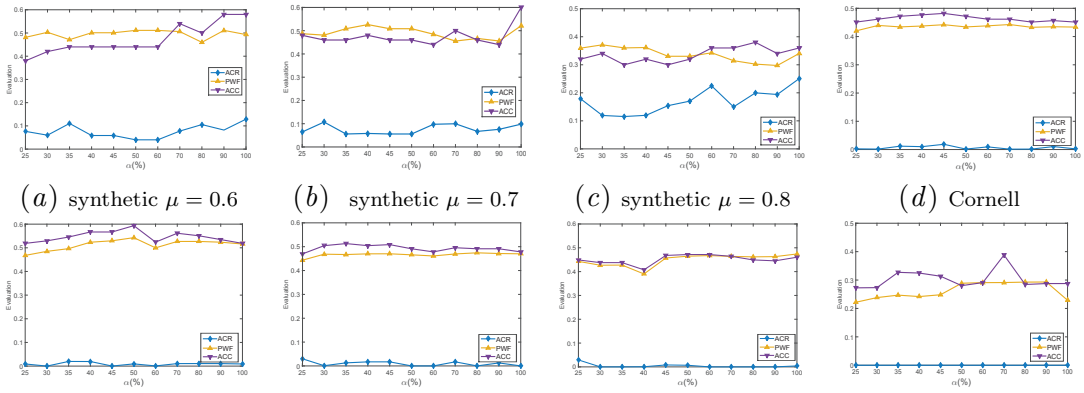
(a) synthetic $\mu = 0.6$ (b) synthetic $\mu = 0.7$ (c) synthetic $\mu = 0.8$ (d) Cornell

(e) Texas (f) Washington (g) Wisconsin (h) Cora

Figure 6: Results *vs.* sample number of possible graphs N.

Besides, we also study the performances of our proposed method upon different values of α, which denotes the rate of edges selected to construct the attribute induced graph. A larger α indicates a lower filtering threshold, thus, more edges are reserved to construct the attribute induced graph. Fig. 7 shows the performances of our method with different setting of α on these four datasets. Generally, according to these figures, the results are not very sensitive to the parameter, and the optimal choice of α is about 0.4 to 0.6. In this study, we set $\alpha = 0.45$. Intuitively, too small α will lead to a sparse graph structure and

result in a worse result, where some informative edges are filtered. In other extreme case, too large α usually incorporates more noise edges, thus it will output non-optimal results. Hence, a value around 0.5 is more tolerant and gets more robust results.



(a) synthetic $\mu = 0.6$ (b) synthetic $\mu = 0.7$ (c) synthetic $\mu = 0.8$ (d) Cornell

(e) Texas (f) Washington (g) Wisconsin (h) Cora

Figure 7: Results *vs.* α in constructing attribute induced graph.

6. Conclusions

In this work, we introduce a new problem of extracting clusters in attributed uncertain graphs involving edge uncertainty and nodes attributes. We aim to use focus attributes as an assistance to refine the graph structure of the attributed uncertain graph by alleviating edge uncertainty. We proposed two novel clustering methods, called AUG-I based on an integrated weighted graph and AUG-U based upon a single unified partition, to extract reliable clusters, where nodes are unlikely to be disconnected and have semantic homogeneity. Extensive studies on real-world datasets demonstrate the effectiveness of our methods in clustering attributed uncertain graphs. Besides, we exhibit the robustness of our methods by parameter analysis on the size of sampling the uncertain graph.

Acknowledgments

This work was supported in part by Beijing Postdoctoral Research Foundation and the National Nature Science Foundation of China (No. 61876016).

References

Charu C Aggarwal and Chandan K Reddy. *Data clustering: algorithms and applications*. CRC Press, 2013.

Pruet Boonma and Juggapong Natwichai. Reliable cluster on uncertain multigraph. In *Proceedings of 18th International Conference on Network-Based Information Systems*, pages 494–498, 2015.

Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Matteo Ceccarello, Carlo Fantozzi, Andrea Pietracaprina, Geppino Pucci, and Fabio Vandin. Clustering uncertain graphs. In *Proceedings of the VLDB Endowment*, pages 472–44. DOI: 10.1145/3164135.3164143, 2017.

Wen-Yen Chen, Yangqiu Song, Hongjie Bai, Chih-Jen Lin, and Edward Y Chang. Parallel spectral clustering in distributed systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(3):568–586, 2011.

Charles J Colbourn. *The Combinatorics of Network Reliability*. New York: Oxford University Press, 1987.

Roger Guimera and Luis A Nunes Amaral. Functional cartography of complex metabolic networks. *Nature*, 433(7028):895–900, 2005.

Stephan Günnemann, Ines Färber, Brigitte Boden, and Thomas Seidl. Subspace clustering meets dense subgraph mining: A synthesis of two paradigms. In *Proceddings of the 10th International Conference Data Mining*, pages 845–850, 2010.

Zahid Halim, Muhammad Waqas, and Syed Fawad Hussain. Clustering large probabilistic graphs using multi-population evolutionary algorithm. *Information Sciences*, 317:78–95, 2015.

Caiyan Jia, Yafang Li, Matthew Carson, Xiaoyang Wang, and Jian Yu. Node attribute-enhanced community detection in complex networks. *Scientific Reports*, 7, 05 2017.

George Kollios, Michalis Potamias, and Evimaria Terzi. Clustering large probabilistic graphs. *IEEE Transactions on Knowledge and Data Engineering*, 25(2):325–336, 2013.

Andrea Lancichinetti, Santo Fortunato, and Filippo Radicchi. Benchmark graphs for testing community detection algorithms. *Physical Review E*, 78(4):046110, 2008.

Laura Langohr and Hannu Toivonen. *Finding representative nodes in probabilistic graphs*. Springer, 2012.

Lin Liu, Ruoming Jin, Charu Aggarwal, and Yelong Shen. Reliable clustering on uncertain graphs. In *Proceddings of the 12th International Conference Data Mining*, pages 459–468. IEEE, 2012.

Flavia Moser, Recep Colak, Arash Rafiey, and Martin Ester. Mining cohesive patterns from graphs with feature vectors. In *Proceedings of International Conference on Data Mining*, pages 593–604, 2009.

Andrew Y Ng, Michael I Jordan, Yair Weiss, et al. On spectral clustering: Analysis and an algorithm. *NIPS*, 2:849–856, 2002.

Bryan Perozzi, Leman Akoglu, Patricia Iglesias Sánchez, and Emmanuel Müller. Focused clustering and outlier detection in large attributed graphs. In *Proceedings of the 20th International Conference on Knowledge Discovery and Data Mining*, pages 1346–1355, 2014.

Dimitrios Rafailidis. Clustering nodes with attributes via graph alignment. In *Proceedings of the 31st Annual ACM Symposium on Applied Computing*, pages 904–907, 2016.

Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. *AI magazine*, 29(3):93, 2008.

Alexander Strehl and Joydeep Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *The Journal of Machine Learning Research*, 3:583–617, 2003.

Eric P Xing, Andrew Y Ng, Michael I Jordan, and Stuart Russell. Distance metric learning with application to clustering with side-information. *NIPS*, 15:505–512, 2003.

Jaewon Yang, Julian J. McAuley, and Jure Leskovec. Community detection in networks with node attributes. In *Proceedings of the 13th International Conference on Data Mining*, pages 1151–1156, 2013.

Tianbao Yang, Rong Jin, Yun Chi, and Shenghuo Zhu. Combining link and content for community detection: a discriminative approach. In *Proceedings of the 15th International Conference on Knowledge Discovery and Data Mining*, pages 927–936, 2009.

Yang Zhou, Hong Cheng, and Jeffrey Xu Yu. Graph clustering based on structural/attribute similarities. In *Proceedings of the VLDB Endowment*, pages 718–729, 2009.