

Overlapping Community Detection in Temporal Text Networks

ABSTRACT

Network is a powerful language to represent relational data. One way to understand network is to analyze groups of nodes which share same properties or functions. The task of discovering such groups is known as *community detection*. Generally, two types of information can be utilized to fulfill this task, i.e., the link structures and the node attributes. The temporal text network is a special kind of network that contains both sources of information. Typical representatives include online blog networks, the World Wide Web (WWW) and academic citation networks. In this paper, we study the problem of overlapping community detection in temporal text network. By examining 32 large temporal text networks, we find a lot of edges connecting two nodes with no common community and discover that nodes in the same community share similar textual contents. This scenario cannot be quantitatively modeled by practically all existing community detection methods. Motivated by these empirical observations, we propose MAGIC (Model Affiliation Graph with Interacting Communities), a generative model which captures community interactions and considers the information from both link structures and node attributes. Our experiments on three categories of datasets show that MAGIC achieves obvious improvements over 4 state-of-the-art methods in terms of 4 widely-used metrics.

CCS Concepts

•Data Mining → Graph Mining; •Networks → Network Clustering;

Keywords

Community Detection, Text Network, Temporal Text Network, Network Analysis

1. INTRODUCTION

Network can serve as a powerful language to represent relational information among data objects from social, natural

and academic domains [33]. One way to understand network is to identify and analyze groups of nodes which share same properties or functions. Such groups of nodes can be users from the same organization in social networks [17], proteins with similar functionality in biochemical networks [10], and papers from the same scientific fields in citation networks [21]. The research task of discovering such groups is known as the *community detection* problem [35]. Traditional methods [17, 22] mainly focus on finding disjoint communities and are all based on a restrictive assumption that each node can only belong to one single community. By relaxing this assumption, the *overlapping community detection* problem becomes more general and has attracted major attention recently [19, 27].

There are generally two types of information that can be utilized to discover overlapping communities [32]. The first one is the *link structure*, i.e., the presence and absence of edges. Classical methods [19, 2, 1] usually focus on this type of information and aim to extract a group of nodes with more links inside the group than between its members and outside the group [18]. The second type of information is the *node attribute*, including online profiles of users, pre-existing features of proteins, and textual contents of papers. Due to the prevalent noise in link structures, the approaches for detecting community based on both types of information [21, 32] have gained increasing popularity.

In this paper, we study the problem of overlapping community detection in temporal text networks. A temporal text network is a directed network in which each node has textual content and temporal information. Such networks are ubiquitous in the real world. Typical representatives include online blog networks, the World Wide Web (WWW), email correspondence networks, and academic citation networks. Identification of meaningful communities in temporal text networks provides useful knowledge for subsequent applications such as domain-specific ranking and user-targeted recommendation.

The contributions of our work are three-folded. First, we gather a collection of 32 temporal text networks with ground-truth communities. They enable us to derive insights into community structures and allow us to quantitatively evaluate community detection methods. Second, we study the interactions among ground-truth communities in temporal text networks and discover that a large proportion of nodes share a link due to community interactions. We also analyze how node attributes help to improve the quality of detected communities and find that nodes in the same communities share similar textual content. Third, based on the

empirical observations, we propose MAGIC (Model Affiliation Graph with Interacting Communities), a probabilistic generative model which utilizes all sources of information in the temporal text network and scales to network with millions of nodes.

Present work: Networks with Ground-Truth Communities. We generate a large set of 32 temporal text networks with reliable ground-truth communities based on Microsoft Academic Graph (MAG)¹ [24]. The MAG dataset contains over 100 million scientific papers with titles, references, publish time, and sets of “Field of Study” (FoS). We construct a temporal text network by sampling an academic citation network for each L1 level FoS under Computer Science (CS) field and define each FoS label as a ground-truth community. Finally, we treat the publish time and title of each paper as its corresponding temporal and textual attributes. More details about data preprocessing are presented in Section 4.1.

Present work: Empirical Observations. The availability of temporal text networks with ground-truth communities enables us to derive insights into the community structure. In this paper, we study the interactions among ground-truth communities and discover that a large number of nodes share a link because of the community interactions. We find that a large proportion of edges connect a pair of nodes which have no communities in common. Current methods [30, 32, 33] fail to identify such community interactions and fail to model this scenario well. We also quantitatively analyze how textual contents provide useful information for overlapping community detection. We find that nodes in the same communities share very similar textual content, which is very intuitive.

Present work: Community Detection in temporal text network. Based on the above empirical observations, we propose MAGIC (Model Affiliation Graph with Interacting Communities), a generative model which models the probability of an edge between two nodes as a function of the communities they share, the interactions among communities they are affiliated in, and the time information of each node. MAGIC captures community interactions and considers the information from both link structures and node attributes. MAGIC further reduces the noise of missing links by utilizing the time information attached on each node. By applying MAGIC to a given temporal text network, we can detect meaningful communities. We conduct extensive experiments on 17 temporal text networks from three different sources – LFR benchmark [11], SNAP, and MAG. Our results show that MAGIC achieves large improvements over 4 state-of-the-art methods [19, 2, 30, 32, 33] in terms of several widely-used metrics [12, 16, 30].

Organization. The rest of this paper is organized as follows. Section 2 summarizes the related work. Section 3 gives formal definitions of temporal text network and the task of overlapping community detection. Section 4 presents our empirical observations. Section 5 introduces the MAGIC along with its learning method. Finally, we report the experimental results in section 6 and conclude in section 7.

2. RELATED WORK

Overlapping community detection has been extensively investigated in the last decade [27]. Classical methods such as

CPM [19], MMSB [2], and LC [1] are mainly based on dense subgraph extraction. For example, CPM aims to find all k -cliques and combine those cliques sharing $k - 1$ nodes to be communities. Consequently, these methods are not applicable for detecting communities in large-scale networks with millions of nodes.

More recently, a series of affiliation graph models [28, 30, 31, 33] are proposed based on the idea that communities arise due to shared group affiliations [5]. Yang and Leskovec introduced Community-Affiliation Graph Model (AGM) [28] in which nodes are affiliated with latent communities they belong to and links are generated based on node community affiliations. They later relaxed the combinatorial optimization problem of fitting AGM and presented a more scalable model called BIGCLAM [30]. This line of works, however, models the underlying affiliation network as a bipartite graph and assumes each community creates edges independently. Compared to these methods, MAGIC relaxes such assumption and captures community interactions.

Another piece of work which also considers community interactions is BNMTF [36]. This method factorizes the adjacency matrix of network into latent factors which are regarded as communities. However, BNMTF keeps using conventional Euclidean distance and generalized KL-divergence as the objective of matrix factorization, which is not scalable and causes bad interpretability.

Many models also study the problem of overlapping community detection in the context of combining link structure with node attributes [32, 21, 15]. A large catalog of such models are based on topic models [23, 34, 3]. However, these methods do not allow a node to have high membership strength in multiple communities simultaneously and therefore leads to unrealistic assumptions about the structure of community overlaps [31]. To solve this problem, authors in [31] proposed CESNA which is an affiliation graph model based on BIGCLAM and used a logistic model to generate binary-valued node attributes. CESNA models the generations of node attributes and link structures as two different mechanisms. Compared to CESNA, MAGIC takes a more unified approach to model these two types of information.

3. PROBLEM FORMULATION

In this section, we formalize the problem of overlapping community detection in temporal text networks. We first define the “text network” and the “temporal text network”. Then, we discuss a method to explicitly encode the text information in graph and define the “projected temporal text network”. Figure 1 illustrates the relationship and the difference among these three types of networks.

Definition 1. (Text Network) A **text network** is defined as a directed unweighted graph $G = (V, E)$, where V is a set of vertices and E is the set of edges between the vertices. Each vertex $v \in V$ represents a document and has a sequence of words associated with it. Each edge $(u, v) \in E$ represents the directed connection between the document u and the document v .

The text network captures the relationship among documents and models it explicitly. Such network is ubiquitous in the real world. Online blog networks, email correspondence networks and academic citation networks are some good representatives.

¹We use the Feb 5, 2016 version of MAG.

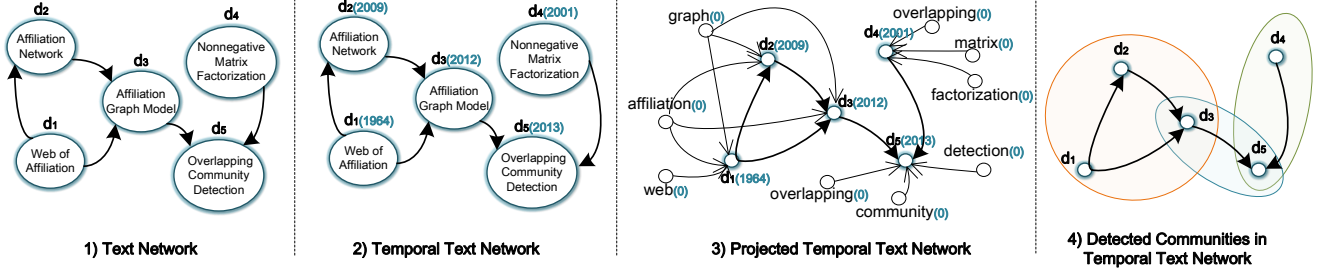


Figure 1: Illustration of three different networks. Blue-colored numbers in parentheses indicate the time information of each node. Clearly, there are two types of information in the temporal text network, i.e., the link structure and the node attribute. The projected temporal text network represents the text information in a more explicit manner and serve as a good proxy for its corresponding temporal text network.

Definition 2. (Temporal Text Network) A **temporal text network** is a text network with time information, denoted as $G = (V, E; T)$. In temporal text network, each vertex $v \in V$ is attached with a timestamp $t(v)$. Furthermore, a temporal text network is called **natural temporal text network** if each edge (u, v) satisfies $t(u) < t(v)$; otherwise, it is called **complex temporal text network**.

The temporal text network encodes the time information of each document. We state that most text networks are natural temporal text networks, provided that we give a proper definition of the edge direction. For example, if we define an edge in citation network starting from the cited paper to the citing one, then this network is natural because nobody can cite future papers.

Definition 3. (Projected Temporal Text Network) A **projected temporal text network**, denoted as $G^P = (V \cup V_w, E \cup E_w; T \cup T_w)$, is a transformation of the original temporal text network $G = (V, E; T)$. Each additional vertex $v \in V_w$ represents a word and each additional edge $(w_i, d_j) \in E_w$ indicates that word w_i exists in document d_j . We set the timestamps of all word vertices to be zero².

Such projection method is proposed in [25] and proved useful to model document-word dependency. The projected temporal text networks serves as a good proxy for the original network. Finally, we state that the problem investigated in this paper is overlapping community detection in temporal text networks. We will later elaborate their differences and discuss how we detect meaningful communities in the temporal text network by exploiting information in its corresponding projected version.

Definition 4. (Overlapping Community Detection in Temporal Text Networks) Given a temporal text network $G = (V, E; T)$, the problem of **overlapping community detection in temporal text network** is to find a collection of subsets of V denoted by $C = \{C_1, \dots, C_K\}$ such that for each $C_i \in C$, its induced subgraph $G[C_i]$ forms a network community³. By allowing $C_i \cap C_j \neq \emptyset$, we can obtain overlapping communities.

²The exact number of this value is actually not important, as long as it is less than the earliest timestamp of all documents.

³An induced subgraph $G[C_i]$ is a graph whose vertex set is C_i and whose edge set consists of all the edges in E that have both endpoints in C_i .

4. EMPIRICAL OBSERVATION

In the section, we first describe how we generate a large collection of temporal text networks and define reliable ground-truth communities. Then, we present our empirical observations by answering two important questions. How many edges connect two nodes that share no common community? How textual contents improve the quality of detected communities? Finally, we discuss the importance of such findings and how they motivate the development of our model.

4.1 Dataset descriptions

We generate temporal text networks with explicit ground-truth communities based on Microsoft Academic Graph (MAG) [24]. The MAG dataset contains over 100 million scientific papers with their titles, references, publish time, and sets of “Field of Study” (FoS) labels. In total, there are over 50 thousands different FoS labels, organized in a four-level hierarchical manner as demonstrated in Figure 2a. Such FoS labels naturally correspond to ground-truth communities since all members (i.e., papers) of the same community are in the same subarea of science and possess the same property. Therefore, we define the FoS labels as the ground-truth communities and further treat the publish time and title of each paper as its temporal and textual attributes.

We construct a temporal text network by sampling an academic citation network. To illustrate the sampling process, we take the “Information Retrieval” (IR) field as an example. We consider that a paper is in IR field if it contains at least one FoS label in the set of *IR-related* FoS labels. A FoS is *IR-related* if it locates in the FoS tree rooted by the “Field of Study” named “Information Retrieval” (IR), as shown in Figure 2a. Then, we construct a citation network among all these selected papers and delete those with no reference and no citation. We repeat this process for 32 L1 level FoS under Computer Science (CS) field. These networks cover a wide range of domains and the sizes of them ranges from thousands of to millions of nodes.

4.2 Empirical Observation

First, we analyze how textual contents help to provide useful information for community detection. For each community, we select two nodes and calculate the Jaccard similarity of their textual contents. The higher this value is, the more similar the textual contents are. We repeat this process for all possible pairs of nodes and get the average Jaccard similarity for that community. We compare this value with the

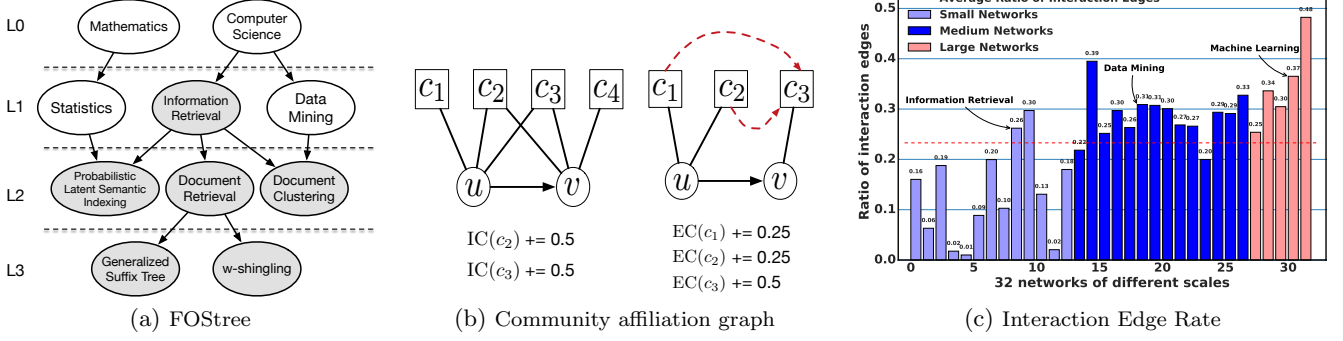


Figure 2: (a) A toy example of Field of Study (FoS) Tree. Each *IR-related* FoS is labeled with a shaded node. (b) An illustrative community affiliation graph. Circles represent nodes in the observed network. Squares represent the latent communities. IC denotes *Internal Connectivity* and EC denotes *External Connectivity*. (c) Ratio of interaction edges in networks with different scales.

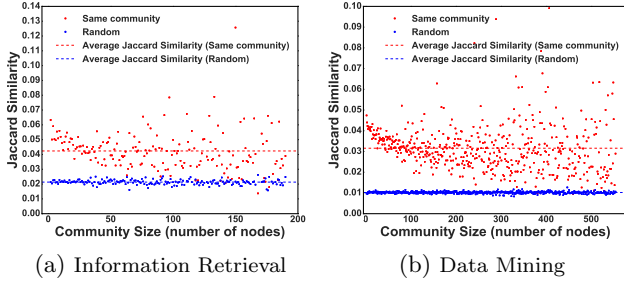


Figure 3: Comparison between the average Jaccard similarity of two randomly selected nodes and that of two nodes from the same community in two temporal text networks.

average Jaccard similarity of a randomly selected set which has the same size of that community. Results are shown in Figure 3. As we can see, the average Jaccard similarity of each community is much higher than that of a randomly selected set. This clearly demonstrates that nodes in the same community have similar textual contents.

Next, we study community interactions by asking the question that how many edges connect two nodes that share no common community? These edges are caused by community interactions and thus we name them *interaction edges*. They reveal the overall amount of community interactions in each dataset. Results are shown in Figure 2c. As we can see, most of networks have more than 20% of edges that are between two nodes with no common community. Besides, the ratio of such edge has an increasing trend with regard to the network size.

We then study such community interactions in a finer granularity. For each edge (u, v) , if node u and v have some communities in common, we assume this edge is generated *only* because two nodes share same communities. On the other hand, if node u and v have no common community, then this edge must be generated by community interactions. We formalize this idea as following. Let $C(u)$ denotes the set of communities of node u . For edge (u, v) , if $C(u) \cap C(v) \neq \emptyset$, then for each community $c \in C(u) \cap C(v)$, we add its *Internal Connectivity* (IC) by $\frac{1}{|C(u) \cap C(v)|}$. Otherwise, if $C(u) \cap C(v) = \emptyset$, then for each community $c \in$

$C(u) \cup C(v)$, we add its *External Connectivity* (EC) by $\frac{1}{2 \times |C(u)|}$ if $c \in C(u)$ and $\frac{1}{2 \times |C(v)|}$ if $c \in C(v)$. An illustrative example is shown in Figure 2b.

After iterating all edges in the network, we can get IC score and EC score of each community. We then define the *Interaction Ratio* of community c as $\frac{EC(c)}{IC(c) + EC(c)}$. Here we ignore all community interactions caused by edges linking two nodes that share some same communities. Take Figure 2b as an example. We ignore possible community interactions between communities c_1 and c_3 or communities c_2 and c_4 . We presume the edge (u, v) is generated *only* due to the *internal connectivity* of community c_2 and c_3 . Therefore, the *Interaction ratio* measures the minimum amount of interaction for each community.

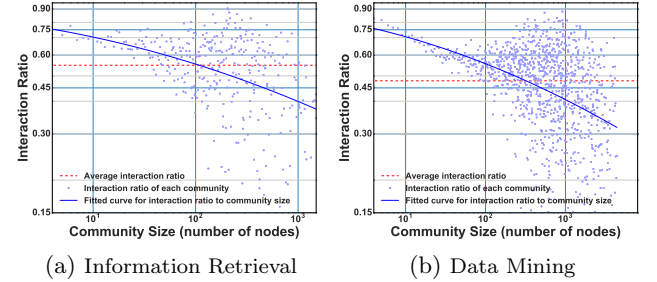


Figure 4: Interaction ratio of each community in four temporal text networks.

As shown in Figure 4, communities have strong interactions. This result is in retrospect, very intuitive. For example, papers in “Information Retrieval” field may adopt the techniques from “Natural Language Processing” papers for semantic search. An algorithm published in a “Machine Learning” conference has its origin from a “Mathematics” problem and been widely used in “Data Mining” field. Consequently, if we find two papers with one in “Data Mining” community and another in “Machine Learning” community, the probability that they share a link should not be modeled as zero, as practically all existing methods do [30, 32, 33]. Instead, we should consider the community interactions and model them explicitly.

5. COMMUNITY DETECTION IN TEMPORAL TEXT NETWORK

Motivated by previous observations, we present MAGIC (Model Affiliated Graph with Interacting Communities), a probabilistic generative model which models the community interactions explicitly.

5.1 Model Description

MAGIC is based on the idea that communities arise due to shared group affiliation [5, 6], and views the whole network as a result generated by a variant of the community-affiliation graph model [28]. Same as the original one, MAGIC models the *community affiliation strength* between each pair of node u and community c with a nonnegative parameter F_{uc} . MAGIC differs mainly in how we model the latent affiliation network. The original community-affiliation graph model treats the affiliation network as a bipartite graph, which fails to capture those important interactions among communities. MAGIC, instead, explicitly models the *community interaction strength* between every pair of community c_i and c_j with a nonnegative parameter η_{ij} ($\eta_{ij} = 0$ indicates community c_i and c_j definitely have no relationship). Finally, we use the parameter η_{ii} to model the probability that two nodes in the same community c_i are connected.

Given those parameters, MAGIC generates a link ($u \rightarrow v$) with the probability $p(u \rightarrow v)$ defined as follows:

$$p(u \rightarrow v) = \left(1 - \exp\left(-\sum_{i,j} F_{ui} \eta_{ij} F_{vj}\right) \right) \delta(u \rightarrow v) \quad (1)$$

$$= \left(1 - \exp(-\mathbf{F}_u^T \boldsymbol{\eta} \mathbf{F}_v) \right) \delta(u \rightarrow v),$$

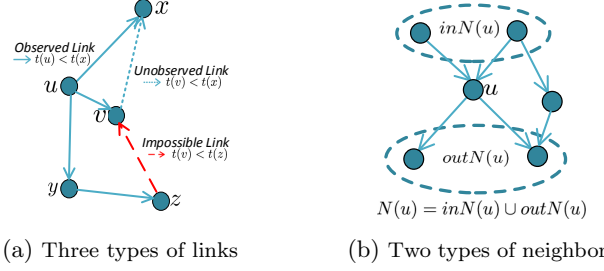
where \mathbf{F}_u is a column vector representing the community affiliation strength for node u , $\boldsymbol{\eta}$ is the community interaction matrix, and $\delta(u \rightarrow v)$ is the weighting function defined only on the timestamps of nodes u and v . The introduction of $\boldsymbol{\eta}$ and δ explicitly models the community interaction and utilizes the node temporal information. In this paper, we mainly focus on the natural temporal text network and thus the weighting function δ is defined as:

$$\delta(u \rightarrow v) = \begin{cases} 1 & \text{if } t(u) < t(v) \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

Eq. 2 essentially restricts the generation of an edge starting from a node with early timestamp and ending with a node with later timestamp. This constraint is ubiquitous in the real world. We cannot cite a paper published in future nor forward an unreceived email.

Next, we discuss how MAGIC utilizes the text information. Instead of treating words and documents separately and use different mechanisms to generate them [32], we adopt a more unified approach. We first construct a projected temporal text network corresponding to the original one and then applied MAGIC to this projected network. A projected temporal text network is intrinsically a heterogeneous network with two types of nodes – “document-node” and “word-node”. MAGIC treats them in the same way and will learn a feature vector representing the latent community affiliation strength for each document and word.

Finally, MAGIC learns the community affiliation matrix \mathbf{F} and the community interaction matrix $\boldsymbol{\eta}$ by maximizing the log likelihood of the observed network G :



(a) Three types of links

(b) Two types of neighbors

Figure 5: (a) Illustration of three types of links. A missing link can be either an *unobserved link* or an *impossible link*. (b) Illustration of two types of neighbors.

$$\hat{\mathbf{F}}, \hat{\boldsymbol{\eta}} = \underset{\mathbf{F} \geq 0, \boldsymbol{\eta} \geq 0}{\operatorname{argmax}} l(\mathbf{F}, \boldsymbol{\eta}), \quad (3)$$

where nonnegative matrices $\mathbf{F} \in \mathbb{R}^{K \times N}$, $\boldsymbol{\eta} \in \mathbb{R}^{K \times K}$ and K , N denote the number of communities and nodes, respectively. The log likelihood can be written out as below:

$$l(\mathbf{F}, \boldsymbol{\eta}) = \sum_{(u \rightarrow v) \in E} \log(1 - \exp(-\mathbf{F}_u^T \boldsymbol{\eta} \mathbf{F}_v)) - \sum_{\substack{(u \rightarrow v) \notin E \\ t(u) < t(v)}} \mathbf{F}_u^T \boldsymbol{\eta} \mathbf{F}_v \quad (4)$$

Notice here we explicitly add the time constraint in the second term so that the absence of an edge (u, v) with $t(u) \geq t(v)$ will not contribute to the likelihood. As demonstrated in Figure 5a, there are two possible reasons for a missing link (u, v) . If we find $t(u) < t(v)$, which means this edge could have been generated, then the absence of such edge can provides some useful information and we define such edge as an *unobserved link*. Otherwise, if $t(u) \geq t(v)$, then the absence of this link carries no information because it cannot be generated anyway. Therefore, we define such edge as an *impossible link*. MAGIC only uses the information derived from *observed links* and *unobserved links*.

5.2 Parameter Learning

We solve the optimization problem defined in Eq. 3 through block coordinate gradient ascent. We first update the community affiliation strength \mathbf{F}_u for each node u with both $\boldsymbol{\eta}$ and \mathbf{F}_v for all other nodes $v \neq u$ fixed, and then update the community interaction matrix $\boldsymbol{\eta}$ with the community affiliation matrix \mathbf{F} fixed. To update the community affiliation strength \mathbf{F}_u for node u , we solve the following subproblem:

$$\hat{\mathbf{F}}_u = \underset{\mathbf{F}_u \geq 0}{\operatorname{argmax}} l(\mathbf{F}_u), \quad (5)$$

where $l(\mathbf{F}_u)$ is the part of $l(\mathbf{F}, \boldsymbol{\eta})$ defined in Eq. 4 that involves \mathbf{F}_u , i.e.,

$$l(\mathbf{F}_u) = \sum_{v \in \text{inN}(u)} \log(1 - \exp(-\mathbf{F}_v^T \boldsymbol{\eta} \mathbf{F}_u)) - \sum_{\substack{v \notin \text{inN}(u) \\ t(v) < t(u)}} \mathbf{F}_v^T \boldsymbol{\eta} \mathbf{F}_u \\ + \sum_{v' \in \text{outN}(u)} \log(1 - \exp(-\mathbf{F}_u^T \boldsymbol{\eta} \mathbf{F}_{v'})) - \sum_{\substack{v' \notin \text{outN}(u) \\ t(v') > t(u)}} \mathbf{F}_u^T \boldsymbol{\eta} \mathbf{F}_{v'}, \quad (6)$$

where $\text{inN}(u)$ and $\text{outN}(u)$ denotes the set of in and out neighbors of node u , respectively. $N(u)$ is equal to $\text{inN}(u) \cup \text{outN}(u)$.

$outN(u)$, as demonstrated in Figure 5b. This subproblem can be further solved by projected gradient ascent [14].

$$F_{uk}^{new} \leftarrow \max\{0, F_{uk}^{old} + \alpha_{F_u} (\nabla l(\mathbf{F}_u))_k\}, \quad (7)$$

where α_{F_u} is the step size computed by backtracking line search [4], and the gradient is:

$$\begin{aligned} \nabla l(\mathbf{F}_u) = & \sum_{v \in inN(u)} \frac{\exp(-\mathbf{F}_v^T \boldsymbol{\eta} \mathbf{F}_u)}{1 - \exp(-\mathbf{F}_v^T \boldsymbol{\eta} \mathbf{F}_u)} \boldsymbol{\eta}^T \mathbf{F}_v - \sum_{\substack{v \notin N(u) \\ t(v) < t(u)}} \boldsymbol{\eta}^T \mathbf{F}_v \\ & + \sum_{v' \in outN(u)} \frac{\exp(-\mathbf{F}_u^T \boldsymbol{\eta} \mathbf{F}_{v'})}{1 - \exp(-\mathbf{F}_u^T \boldsymbol{\eta} \mathbf{F}_{v'})} \boldsymbol{\eta} \mathbf{F}_{v'} - \sum_{\substack{v' \notin N(u) \\ t(v') > t(u)}} \boldsymbol{\eta} \mathbf{F}_{v'}. \end{aligned} \quad (8)$$

After the parameter \mathbf{F} is updated, we fix \mathbf{F} and update the community interaction matrix $\boldsymbol{\eta}$. Notice that $\boldsymbol{\eta}$ is involved in every term of Eq. 4 and thus we solve it directly.

$$\boldsymbol{\eta}_{ij}^{new} \leftarrow \max\{0, \boldsymbol{\eta}_{ij}^{old} + \alpha_{\boldsymbol{\eta}} (\nabla_{\boldsymbol{\eta}} l(\mathbf{F}, \boldsymbol{\eta}))_{ij}\}, \quad (9)$$

where the step size $\alpha_{\boldsymbol{\eta}}$ is also calculated by backtracking line search, and the gradient for $\boldsymbol{\eta}$ is:

$$\nabla_{\boldsymbol{\eta}} l(\mathbf{F}, \boldsymbol{\eta}) = \sum_{(u \rightarrow v) \in E} \frac{\exp(-\mathbf{F}_u^T \boldsymbol{\eta} \mathbf{F}_v)}{1 - \exp(-\mathbf{F}_u^T \boldsymbol{\eta} \mathbf{F}_v)} \mathbf{F}_u \mathbf{F}_v^T - \sum_{\substack{(u \rightarrow v) \notin E \\ t(u) < t(v)}} \mathbf{F}_u \mathbf{F}_v^T. \quad (10)$$

We notice from Eqs. (8) and (10) that direct computations of $\nabla l(\mathbf{F}_u)$ and $\nabla_{\boldsymbol{\eta}} l(\mathbf{F}, \boldsymbol{\eta})$ take $O(N)$ and $O(N^2)$ time, respectively. To reduce the time complexity and increase scalability, we adopt the following tricks:

$$\sum_{\substack{v \notin N(u) \\ t(v) < t(u)}} \boldsymbol{\eta}^T \mathbf{F}_v = \sum_{t(v) < t(u)} \boldsymbol{\eta}^T \mathbf{F}_v - \sum_{v \in inN(u)} \boldsymbol{\eta}^T \mathbf{F}_v, \quad (11)$$

$$\sum_{\substack{v' \notin N(u) \\ t(v') > t(u)}} \boldsymbol{\eta} \mathbf{F}_{v'} = \sum_{t(v') > t(u)} \boldsymbol{\eta} \mathbf{F}_{v'} - \sum_{v' \in outN(u)} \boldsymbol{\eta} \mathbf{F}_{v'}, \quad (12)$$

$$\sum_{\substack{(u \rightarrow v) \notin E \\ t(u) < t(v)}} \mathbf{F}_u \mathbf{F}_v^T = \sum_{\substack{(u \rightarrow v) \\ t(u) < t(v)}} \mathbf{F}_u \mathbf{F}_v^T - \sum_{\substack{(u \rightarrow v) \in E \\ t(u) < t(v)}} \mathbf{F}_u \mathbf{F}_v^T. \quad (13)$$

In this way, we can compute $\nabla l(\mathbf{F}_u)$ in $O(N(u))$ by caching the first term in the right hand side of Eqs. (11) and (12), and compute $\nabla_{\boldsymbol{\eta}} l(\mathbf{F}, \boldsymbol{\eta})$ in $O(|E|)$ by caching the first term in the right hand side of Eq. 13. We notice that the combined time complexity for updating the whole \mathbf{F} is $O(|E|)$. Therefore, we conclude that the time complexity of each iteration for MAGIC is $O(|E|)$.

5.3 Other Issues

Model Initialization. To initialize \mathbf{F} , we extend the method in [8] to directed network. The conductance in directed network is also defined in [7]. The in-neighbors $inN(u)$ of node u is *locally minimal* if $inN(u)$ has lower conductance than all in-neighbors $inN(v)$ where node $v \in outN(u)$. For a node u' belonging to such a locally minimal neighborhood k , we initialize $F_{u'k} = 1$, otherwise we let $F_{u'k} = 0$. To initialize $\boldsymbol{\eta}$, we set the entries in the main diagonal as 0.9 and all other entries to be 0.1.

Determining community membership. After learning parameters $\hat{\mathbf{F}}$ and $\hat{\boldsymbol{\eta}}$, we need to determine the “hard” community membership of each node. We achieve this by

thresholding $\hat{\mathbf{F}}$ with a set of $\{\delta_k\}$, one for each community c_k . The basic intuition is that if two nodes belong to the same community c_k , then the probability of having an link between them through community c_k is larger than $1/N$, where N is the number of nodes. Following this idea, we can obtain δ_k as below:

$$\delta_k = \sqrt{-\frac{\log(1 - 1/N)}{\eta_{kk}}}. \quad (14)$$

With $\{\delta_k\}$ obtained, we consider node u belonging to community k if $F_{uk} \geq \delta_k$.

Choosing the number of communities. We use the method in [2] to choose the number of communities K . Specifically, we reserve 20% of links for validation and learn the model parameters with the remaining 80% of links for different K . After that, we use the learned parameters to predict the links in validation set and select the K with the maximum prediction score as the number of communities.

6. EXPERIMENTS

6.1 Experiment Setup

6.1.1 Dataset

We evaluate our model using three categories of networks with ground-truth communities. In all networks, each node is assigned to at least one community and thus all quantitative metrics are applicable.

MAG (Microsoft Academic Graph) We construct 32 temporal text network based on the MAG as described in the Section 4, from which we randomly choose 11 for the experiment. The source codes and cleaned datasets are available online⁴.

SNAP (Standard Network Analysis Project) SNAP provides six networks with ground-truth communities, among which we choose three, the Amazon product co-purchasing network, the Youtube social network and the LiveJournal blogging community⁵. For the experiment, we create 500 different subnetworks for each of the two datasets as in [30].

LFR (Lancichinetti-Fortunato-Radicchi) Benchmark LFR [11] is one of the state-of-the-art synthetic networks with overlapping ground-truth communities. In our experiments, we change parameters N-D-R to generate three different networks, named LFR Small/Middle/Large. Here N is the number of nodes in the network, D is the average degree and R is the ratio of nodes which belong to multiple communities. For the LFR Small/Middle/Large, we set N-D-R as (10k-8-0.4), (50k-12-0.6), (100k-16-0.8) respectively.

6.1.2 Comparative Methods

We compare MAGIC with another 4 baseline methods: Clique Percolation Method (CPM) [19], Mixed-Membership Stochastic Block Model (MMSB) [2], Cluster Affiliation Model for Big Networks (BIGCLAM) [30], and Communities from Edge Structure and Node Attributes (CSENA). CPM and MMSB are representatives of methods based on dense subgraph extraction while BIGCLAM and CSENA are representatives of methods based on affiliation graph model. For

⁴<http://tinyurl.com/WSDM2017-205>

⁵<http://snap.stanford.edu>

Metric	Method	MAG Internet Privacy	MAG Computer Hardware	MAG Information Retrieval	LFR Small	LFR Middle	LFR Large	SNAP Amazon	SNAP Youtube	SNAP Live Journal
Coverage ratio	MAGIC	*0.998	*1.000	*1.000	*0.999	*1.000	*0.999	*0.996	0.832	0.972
	CPM	0.271	0.017	0.047	0.053	0.019	0.008	0.813	0.233	0.939
	MMSB	0.823	0.799	0.799	0.908	0.967	0.984	0.958	*0.836	*0.983
	BIGCLAM	0.861	0.561	0.380	0.611	0.409	0.307	0.982	0.671	0.953
	CESNA	0.875	0.716	0.838	-	-	-	-	-	-
F1 score	MAGIC	*0.235	*0.175	*0.171	*0.245	*0.061	*0.019	*0.173	*0.070	0.063
	CPM	0.137	0.008	0.021	0.050	0.008	0.005	0.094	0.011	0.056
	MMSB	0.199	0.053	0.087	0.147	0.030	0.011	0.153	0.046	0.060
	BIGCLAM	0.208	0.104	0.073	0.157	0.036	0.019	0.168	0.055	*0.071
	CESNA	0.210	0.107	0.121	-	-	-	-	-	-
Modularity	MAGIC	0.506	0.500	0.498	*0.500	*0.500	*0.500	*0.614	*0.543	0.665
	CPM	0.411	0.436	0.418	0.337	0.349	0.263	0.607	0.233	*0.845
	MMSB	0.518	0.501	0.500	*0.500	*0.500	*0.500	0.581	0.518	0.682
	BIGCLAM	0.538	*0.561	*0.550	*0.500	*0.500	*0.500	0.570	0.522	0.642
	CESNA	*0.542	0.560	0.548	-	-	-	-	-	-
Omega index	MAGIC	*0.426	*0.406	0.508	0.886	0.797	0.683	0.389	*0.422	0.120
	CPM	0.388	0.336	0.559	*0.969	*0.987	*0.989	0.277	0.132	0.080
	MMSB	0.392	0.350	0.561	0.896	0.819	0.750	0.401	0.403	0.163
	BIGCLAM	0.395	0.347	0.562	0.946	0.965	0.971	*0.460	0.375	*0.201
	CESNA	0.397	0.350	*0.564	-	-	-	-	-	-

Table 1: Results of 5 methods on 9 networks from 3 different categories.

CPM, we set the clique size $k = 4$ and use the implementation in the Stanford Network Analysis Platform (SNAP)⁶. In addition to CPM, SNAP also provides implementation of BIGCLAM and CESNA and we adopt them for the experiment. For MMSB, We use the implementation in [9]⁷.

Previously BIGCLAM has been shown to outperform NMF [20] [26] and CSENA has been shown to outperform CODICIL [21] and Block-LDA [3]. Therefore, we do not compare with those algorithms.

6.1.3 Metrics

We denote the set of ground-truth communities as C and the set of detected communities as \hat{C} . To measure the performance of our model, we select following 4 metrics.

Coverage ratio is the ratio of nodes which can be assigned to at least one community by the model. Intuitively, a model cannot be useful if it can only detects communities for a very few proportion of nodes.

F1 score is the average of the F1 score of the best-matching ground-truth community to reach detected community. Please refer to [30] for details.

Modularity is an unsupervised metric for communities. We use the extended version of the modularity to apply it on the overlapping communities. Please refer to [13] for details.

Omega index is a metric with the basic idea of estimating the number of communities that each pair of nodes shares. Please refer to [30] for details.

For all 4 metrics higher values mean that the detected communities are more accurate and have better qualities.

6.2 Quantitative Results

6.2.1 Performance on three categories of network

Table 1 shows the performance of 5 methods on 9 networks in terms of 4 metrics. These 9 networks belong to 3 different categories. And only MAG networks contain textual content so that for CESNA we only measure it on MAG. In addition, SNAP networks are undirected networks so that for model

MAGIC we use MAGIC(all) for MAG, MAGIC(net) for LFR and MAGIC(raw) for SNAP and all of these versions are referred to as MAGIC in the table.

First, we notice that the coverage ratio of MAGIC in most networks is greater than 0.996, which means it can label almost all nodes with at least one community. In addition, the F1 score of MAGIC is the highest compared to other methods except in one network. The remaining two metrics are bias to models without considering community interaction for the reason that they only focus on edges or nodes in the same community. But for our model, even though two nodes belong to different communities, it's also possible for them to be linked because of community interaction. Due to this property of the two metrics, MAGIC doesn't get the highest score.

6.2.2 Scaled performance comparison

According to Table 1, we can see different metrics measure different aspects community detection algorithms and it's hard to find a method that totally outperforms others for all metrics. To compare the overall performance of a method, we need to normalize those metrics to a same scale. Specifically, we scale it to make sure the best community detection method will get a score 1. Then, we sum up all four normalized scores to obtain the composite score. In order to run the integrated MAGIC, i.e., MAGIC(all), we only randomly choose 7 networks from MAG datasets which contain temporal and textual information.

Figure 6 compares the performance of 6 methods on 7 MAG networks in terms of the sum of four normalized metrics. We can see that MAGIC(all) achieves the best performance in all the 7 networks and mostly MAGIC(net) gets the second place. The average composite performance of MAGIC(all) is 3.68, which is 41% higher than BIGCLAM(2.61), 19% higher than CESNA(3.11), 138% higher than CPM(1.55), and 33% higher than MMSB(2.88).

6.3 Effects of Community Interactions

We further analyze how community interactions affect quality of detected communities when different combinations of information sources are used. To achieve this, we intro-

⁶<https://github.com/snap-stanford/snap>

⁷MMSB: <https://github.com/premgopalan/svinet>

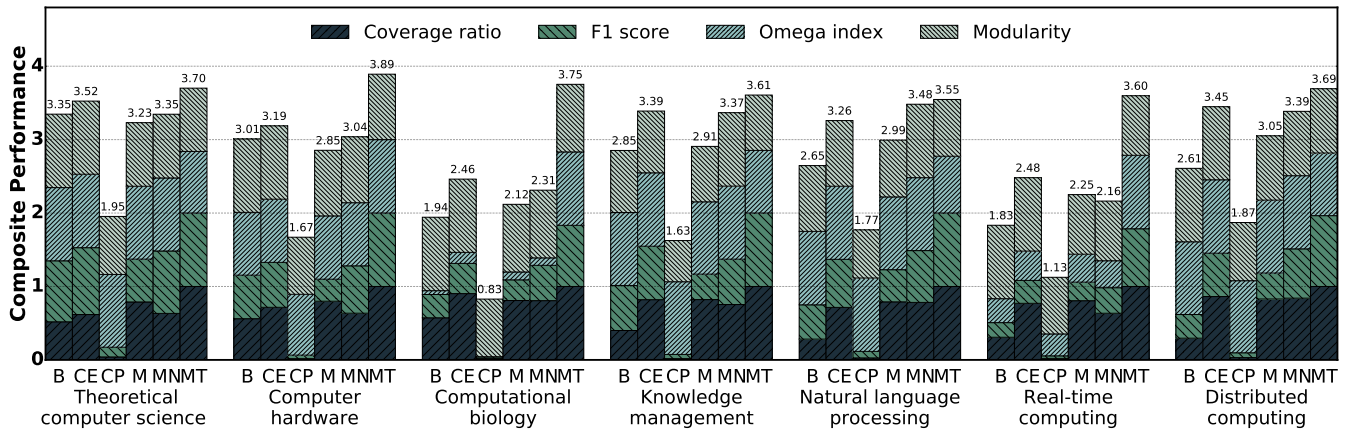


Figure 6: The composite performance of 6 methods on 7 small scale temporal text networks. B: BIGCLAM; CE: CESNA, CP: CPM, M: MMSB, MN: MAGIC(net), MA: MAGIC(all).

Method	Directed	Text	Community Interaction	MAG Computer graphics(images)	MAG Knowledge management	MAG Information retrieval	LFR Small	LFR Large	SNAP amazon	SNAP youtube
BIGCLAM	F	F	F	3.042	2.853	2.702	3.414	3.291	*3.885	3.430
MAGIC(raw)	F	F	T	*3.117	*3.109	*3.219	*3.958	*3.720	3.846	*3.995
CoDA	T	F	F	3.145	3.128	3.246	3.499	3.212	-	-
MAGIC(net)	T	F	T	*3.463	*3.607	*3.351	*3.913	*3.690	-	-
CESNA	F	T	F	3.410	3.389	3.377	-	-	-	-
MAGIC(all)	F	T	T	*3.533	*3.607	*3.574	-	-	-	-

Table 2: Results of six methods on 7 networks of 3 different categories. ‘T’ denotes we consider a certain type of information and ‘F’ means we do not. We can see the introduction of community interactions clearly improves the performance, when different combinations of information sources are provided.

duce a new method called CoDA (Communities through Directed Affiliations) [33], an overlapping community detection method that applies to directed networks. Totally, we have six methods to compare and we divide them into three groups. Table 2 shows the composite performance of these methods. Since the Amazon network and the Youtube network are undirected and without node attributes, only BIGCLAM and MAGIC(raw) are able to run on them. Likewise, CESNA and MAGIC(all) can not process the LFR networks due to the absence of node attributes in the synthetic networks.

In the first group, we ignore all node attributes (both temporal information and textual contents) as well as edge directions, and compare the results of BIGCLAM and MAGIC(raw). We can see MAGIC(raw) beats BIGCLAM in six out of seven networks. The average performance of MAGIC(raw) is 3.57, which is about 10% higher than that of BIGCLAM (3.23). Such improvements occur in another two groups where we use the edge directions and text information, respectively. We contribute these improvements to the introduction of community interactions as the only variable in all three groups is whether such interactions are considered or not.

6.4 Word Community

Using MAGIC, we can actually obtain both document and word communities for each temporal text network. Those word communities have semantic meanings. Here, we present two example word communities from the “Theoretical Computer Science” field. We choose the field “Theoretical computer science” as it covers a broad range of subjects including “Computational biology”, “Computer network”, and etc.

Meaningful word communities can be observed in Figure 7.



Figure 7: Word Clouds of Word Communities

7. CONCLUSION

In this paper, we study the problem of overlapping community detection in temporal text networks. We generate a large set of 32 temporal text networks with reliable ground-truth communities. They enable us to quantitatively study the community structure and evaluate community detection methods. We study the interactions among communities and discover that many nodes share a link due to such community interactions. We also find that nodes in the same community have similar textual contents. Based on these empirical observations, we propose MAGIC, a generative model which explicitly models the community interactions and utilizes the information from both link structures and node attributes. Extensive experiments based on 17 networks from 3 different sources demonstrate the effectiveness of MAGIC.

8. REFERENCES

- [1] Y.-Y. Ahn, J. P. Bagrow, and S. Lehmann. Link communities reveal multiscale complexity in networks. *Nature*, 466(7307):761–764, 2010.
- [2] E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing. Mixed membership stochastic blockmodels. In *Advances in Neural Information Processing Systems*, pages 33–40, 2009.
- [3] R. Balasubramanian and W. W. Cohen. Block-lda: Jointly modeling entity-annotated text and entity-entity links. In *SDM*, volume 11, pages 450–461. SIAM, 2011.
- [4] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.
- [5] R. L. Breiger. The duality of persons and groups. *Social forces*, 53(2):181–190, 1974.
- [6] S. Fortunato and M. Barthélemy. Resolution limit in community detection. *Proceedings of the National Academy of Sciences*, 104(1):36–41, 2007.
- [7] S. Galhotra, A. Bagchi, S. Bedathur, M. Ramanath, and V. Jain. Tracking the conductance of rapidly evolving topic-subgraphs. *Proc. VLDB Endow.*, 8(13):2170–2181, Sept. 2015.
- [8] D. F. Gleich and C. Seshadhri. Vertex neighborhoods, low conductance cuts, and good seeds for local community methods. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 597–605. ACM, 2012.
- [9] P. K. Gopalan and D. M. Blei. Efficient discovery of overlapping communities in massive networks. *Proceedings of the National Academy of Sciences*, 110(36):14534–14539, 2013.
- [10] N. J. Krogan, G. Cagney, H. Yu, G. Zhong, X. Guo, A. Ignatchenko, J. Li, S. Pu, N. Datta, A. P. Tikuisis, et al. Global landscape of protein complexes in the yeast *saccharomyces cerevisiae*. *Nature*, 440(7084):637–643, 2006.
- [11] A. Lancichinetti and S. Fortunato. Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *PHYSICAL REVIEW E Phys Rev E*, 80:016118, 2009.
- [12] A. Lancichinetti, S. Fortunato, and J. Kertész. Detecting the overlapping and hierarchical community structure in complex networks. *New Journal of Physics*, 11(3):033015, 2009.
- [13] A. Lazar, D. Abel, and T. Vicsek. Modularity measure of networks with overlapping communities, 2009. cite arxiv:0910.5072.
- [14] C.-J. Lin. Projected gradient methods for nonnegative matrix factorization. *Neural computation*, 19(10):2756–2779, 2007.
- [15] L. Liu, L. Xu, Z. Wangy, and E. Chen. Community detection based on structure and content: A content propagation perspective. In *Data Mining (ICDM), 2015 IEEE International Conference on*, pages 271–280, Nov 2015.
- [16] A. F. McDaid, D. Greene, and N. Hurley. Normalized mutual information to evaluate overlapping community finding algorithms. *arXiv preprint arXiv:1110.2515*, 2011.
- [17] M. E. Newman. Detecting community structure in networks. *The European Physical Journal B-Condensed Matter and Complex Systems*, 38(2):321–330, 2004.
- [18] M. E. Newman. Modularity and community structure in networks. *Proceedings of the national academy of sciences*, 103(23):8577–8582, 2006.
- [19] G. Palla, I. Derényi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814–818, 2005.
- [20] I. Psorakis, S. Roberts, M. Ebden, and B. Sheldon. Overlapping community detection using bayesian non-negative matrix factorization. *Physical Review E*, 83(6):066114, 2011.
- [21] Y. Ruan, D. Fuhry, and S. Parthasarathy. Efficient community detection in large networks using content and links. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1089–1098. International World Wide Web Conferences Steering Committee, 2013.
- [22] S. E. Schaeffer. Graph clustering. *Computer Science Review*, 1(1):27–64, 2007.
- [23] J. Shen, Z. Song, S. Li, Z. Tan, Y. Mao, L. Fu, L. Song, and X. Wang. Modeling topic-level academic influence in scientific literatures. In *Workshops at the Thirtieth AAAI Conference on Artificial Intelligence*, 2016.
- [24] A. Sinha, Z. Shen, Y. Song, H. Ma, D. Eide, B.-J. P. Hsu, and K. Wang. An overview of microsoft academic service (mas) and applications. In *Proceedings of the 24th International Conference on World Wide Web, WWW '15 Companion*, pages 243–246, New York, NY, USA, 2015. ACM.
- [25] J. Tang, M. Qu, and Q. Mei. Pte: Predictive text embedding through large-scale heterogeneous text networks. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1165–1174. ACM, 2015.
- [26] F. Wang, T. Li, X. Wang, S. Zhu, and C. Ding. Community discovery using nonnegative matrix factorization. *Data Mining and Knowledge Discovery*, 22(3):493–521, 2011.
- [27] J. Xie, S. Kelley, and B. K. Szymanski. Overlapping community detection in networks: The state-of-the-art and comparative study. *Acm computing surveys (csur)*, 45(4):43, 2013.
- [28] J. Yang and J. Leskovec. Community-affiliation graph model for overlapping network community detection. In *Data Mining (ICDM), 2012 IEEE 12th International Conference on*, pages 1170–1175. IEEE, 2012.
- [29] J. Yang and J. Leskovec. Defining and evaluating network communities based on ground-truth. In *Proceedings of the ACM SIGKDD Workshop on Mining Data Semantics, MDS '12*, pages 3:1–3:8, New York, NY, USA, 2012. ACM.
- [30] J. Yang and J. Leskovec. Overlapping community detection at scale: a nonnegative matrix factorization approach. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 587–596. ACM, 2013.
- [31] J. Yang and J. Leskovec. Structure and overlaps of ground-truth communities in networks. *ACM Trans. Intell. Syst. Technol.*, 5(2):26:1–26:35, Apr. 2014.
- [32] J. Yang, J. McAuley, and J. Leskovec. Community detection in networks with node attributes. In *Data mining (ICDM), 2013 IEEE 13th international conference on*, pages 1151–1156. IEEE, 2013.
- [33] J. Yang, J. McAuley, and J. Leskovec. Detecting cohesive and 2-mode communities in directed and undirected networks. In *Proceedings of the 7th ACM international conference on Web search and data mining*, pages 323–332. ACM, 2014.
- [34] T. Yang, R. Jin, Y. Chi, and S. Zhu. Combining link and content for community detection: a discriminative approach. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 927–936. ACM, 2009.
- [35] H. Zhang, I. King, and M. R. Lyu. Incorporating implicit link preference into overlapping community detection. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, AAAI'15*, pages 396–402. AAAI Press, 2015.
- [36] Y. Zhang and D.-Y. Yeung. Overlapping community detection via bounded nonnegative matrix tri-factorization. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 606–614. ACM, 2012.