Exponentially Twisted Sampling for Centrality Analysis and Community Detection in Attributed Networks

Cheng-Hsun Chang, Cheng-Shang Chang, *Fellow, IEEE,* Chia-Tai Chang, Duan-Shin Lee, *Senior Member, IEEE*, and Ping-En Lu

Abstract—In this paper, we conduct centrality analysis and community detection for attributed networks. An attributed network, as a generalization of a graph, has node attributes and edge attributes that represent the “features” of nodes and edges. Traditionally, centrality analysis and community detection of a graph are done by providing a sampling method, such as a random walk, for the graph. To take node attributes and edge attributes into account, the sampling method in an attributed network needs to be twisted from the original sampling method in the underlining graph. For this, we consider the family of exponentially twisted sampling methods and propose using path measures to specify how the sampling method should be twisted. For signed networks, we define the influence centralities by using a path measure from opinions dynamics and the trust centralities by using a path measure from a chain of trust. For attributed networks with node attributes, we also define advertisement-specific influence centralities by using a specific path measure that models influence cascades in such networks. For networks with a distance measure, we define the path measure as the total distance along a path. By specifying the desired average distance between two randomly sampled nodes, we are able to detect communities with various resolution parameters. Various experiments are conducted to further illustrate these exponentially twisted sampling methods by using three real datasets: the political blogs, the MemeTracker dataset, and the WonderNetwork.

Index Terms—Centralities, communities, signed networks, exponentially twisted sampling.

1 Introduction

Centrality analysis [1], [2] and community detection [3], [4] have been two of the most important research topics in social network analysis. In the literature (see e.g., the book [5]), there are various notions of centralities defined for ranking the importance of nodes in a network, including the degree centrality, eigenvector centrality, Katz centrality, PageRank, closeness centrality, and betweenness centrality. Among them, PageRank [6], proposed by Google, is perhaps one of the most famous centrality measures for ranking web pages. The key idea behind PageRank is to model the behavior of a web surfer by a random walk (the random surfer model) and then use that to sample the probability for a web surfer to visit a specific web page. The higher the probability for a web surfer to visit a specific web page is, the more important that web page is. Personalized PageRank [7], [8] is a generalization of PageRank. In the original PageRank, the random surfer starts a new web page *uniformly* among all the web pages. The key insight of Personalized PageRank is to use a *biased* selection to represent personal interest among all the web pages, and that leads a different sampling probability for a web surfer to visit a specific web page.

The basic goal of *community detection* is to find a partition of a graph so as to discover and understand the large-scale structure of a network. As commented in [9], researchers from different fields have different opinions on what a *good community* should look like. There are several notions for this in the literature: (i) a good community should have more edges within the community than the edges going outside the community (see e.g., [10], [11]), (ii) a good community should be densely connected [12], [13], [14], (iii) a graph with a good community structure should behave quite differently from random graphs [15], [16], (iv) a good community should have a high probability to trap a random walker inside the community [17], [18], [19], [20], and (v) rumors are spread fast within a good community [21]. As mentioned before, people have different views. As such, community detection (clustering) is in general considered as an ill-posed problem [22], and communities can only be formally defined on top of a specific viewpoint [9], [23]. To obtain a viewpoint of a network, one typical method is to “sample” the network, e.g., edge sampling, random walks, diffusion [16], or random gossiping [21]. Mathematically, each sampling method renders a (probability) measure for further analysis of the sampled network. The goal of this paper is to provide a general framework for centrality analysis and community detection in attributed networks. We do not aim to solve a specific ranking task or a community detection task for a specific dataset.

In this paper, we consider attributed networks. An attributed network is a generalization of a network (graph). In addition to the set of nodes and the set of edges in an underlining network, an attributed network could have node attributes and edge attributes that specify the “features” of nodes and edges. Our study of attributed networks is moti-

- *C.-H. Chang, C.-S. Chang, C.-T. Chang, D.-S. Lee, and P.-E. Lu are with the Institute of Communications Engineering, National Tsing Hua University, Hsinchu 300, Taiwan, R.O.C.*
  E-mail: hhh42011@gmail.com, cschang@ee.nthu.edu.tw, e158420502@hotmail.com, lds@cs.nthu.edu.tw, j94223@gmail.com.

The conference version of this paper was accepted for presentation in ICC 2018.

vated by the recent advances of online social networks. Online social networks contain more information than merely nodes (persons) and edges (interactions), e.g., the personal information of a person, the dialogues between two persons, the comments/interests/ratings of a person on various topics, et al. Such side information can be further processed to gain more insights and improve solutions to classic tasks of analyzing online social networks (see e.g., [24], [25], [26], [27], [28], [29], [30]). For instance, the dialogues between two persons can be used for knowing whether the two persons are friends or enemies. In particular, as pointed out in [25], users on Epinions can express trust or distrust of others, and users on Slashdot can declare others to be either "friends" or "foes." That leads to a special class of attributed networks, called *signed networks*, where each edge in a signed network is labelled with either a positive sign or a negative sign to indicate the friendship/enemy or trust/distrust relationship between the two ends of an edge. Another typical example of an attributed network is that every node in the network represents a person with different ratings/interests on various topics [26], [29]. The problem we would like to address is how we rank nodes in attributed networks. Such a problem will be called the centrality analysis in attributed networks. Another example of an attributed network is a network with a distance measure that specifies the distance between two nodes. We will also address how community detection can be done in such an attributed network.

Our approach to centrality analysis and community detection in attributed networks is to use the probabilistic framework for structural analysis in networks in [9], [23], [31]. Like PageRank [6], the probabilistic framework requires a sampling method of a network, called a *viewpoint*. The sampling method of a network is characterized by a probability measure for randomly selecting a path r in the network. In order to take the attributes of nodes and edges into account, one needs a *biased* viewpoint as previously discussed in Personalized PageRank [7], [8]. As such, the sampling method in an attributed network (and the corresponding probability measure) needs to be a *twisted* probability measure of its underlining network. For this, we propose using a path measure $f(r)$ that maps every path r in an attributed network to a real-valued vector. By specifying the average values of a path measure, we then have a set of constraints for the twisted sampling probability measure. This then leads to the exponentially twisted probability measure [32], [33], [34] that minimizes the Kullback-Leibler distance between the twisted probability measure and the original probability measure under the set of constraints from the average values of the path measure.

Each path measure with specified average values leads to a method of ranking nodes in an attributed network and that method is in general different from the original ranking method in its underlining network. For centrality analysis, we introduce three path measures in attributed networks and that leads to three new notions of centralities in attributed networks. For signed networks that have both positive edges and negative edges, we show how the influence centralities can be defined by using a path measure derived from opinions dynamics and how the trust centralities can be defined by using a path measure derived from a chain of trust. In particular, we show that one may vary the specified average value of the path measure in a signed network so that the influence centrality is turned into the positive degree ranking, the negative degree ranking, and the total degree ranking. For attributed networks with node attributes, we also show how advertisement-specific influence centralities can be defined by using a specific path measure that models influence cascades in such networks. For community detection, we consider an attributed network with a distance measure. We show communities with different sizes can be detected by specifying a resolution parameter derived from the average distance between two randomly sampled nodes. We then conduct various experiments to illustrate these centralities by using two real datasets: the political blogs in [35] and the MemeTracker dataset [36]. For the political blogs, our numerical results show how the top 100 nodes are changed when the specified average value of the path measure for influence centrality in a signed network is changed. For the MemeTracker dataset, our numerical results show how the top 100 nodes are changed when the topics of the advertisement-specific influence centralities are changed. To illustrate the effect of the resolution parameter for the community detection problem, we also conduct various experiments by using a synthetic dataset and the real network from the WonderNetwork website [37].

The rest of the paper is organized as follows. In Section 2, we review the probabilistic framework of sampled graphs in undirected/directed networks. We then generalize such a probabilistic framework to attributed networks by using exponentially twisted sampling in Section 3. In Section 4, we introduce various path measures for centrality analysis and community detection in various attributed networks. We then conduct various experiments to evaluate the effects of these path measures in Section 5. The paper is concluded in Section 6.

2 Review of the probabilistic framework of sampled graphs

In [9], [23], a probabilistic framework for structural analysis in undirected/directed networks was proposed. The main idea in that framework is to sample a network by randomly selecting a path in the network. A network with a path sampling distribution is then called a *sampled graph* in [9], [23] that can, in turn, be used for structural analysis of the network, including centrality and community. Specifically, suppose a network is modeled by a graph $G(V, E)$, where V denotes the set of vertices (nodes) in the graph and E denotes the set of edges (links) in the graph. Let $n = |V|$ be the number of vertices in the graph and index the n vertices from $1, 2, \dots, n$. Also, let $A = (a_{i,j})$ be the $n \times n$ adjacency matrix of the graph, i.e.,

$$a_{i,j} = \begin{cases} 1, & \text{if there is an edge from vertex } i \text{ to vertex } j, \\ 0, & \text{otherwise.} \end{cases}$$

Let $R_{u,w}$ be the set of (directed) paths from u to w and $R = \cup_{u,w \in V} R_{u,w}$ be the set of paths in the graph $G(V, E)$. According to a probability mass function $p(\cdot)$, called the *path sampling distribution*, a path $r \in R$ is selected at random with probability $p(r)$. In [9], [23], there are many methods for sampling a graph with a randomly selected path. Here we introduce the following three commonly used approaches: (i) sampling by uniformly selecting a directed edge, (ii) sampling by a Markov chain, and (iii) sampling by a random walk on an undirected network with path length 1 or 2.

Example 1. **(Sampling by uniformly selecting a directed edge)** Given a directed graph $G = (V, E)$ with the adjacency matrix $A = (a_{i,j})$, one only sample directed paths with length 1 and this is done by uniformly selecting a directed edge among all the directed edges. Specifically, sampling by uniformly selecting a directed edge has the following probability mass function:

$$p(r) = \begin{cases} 1/m, & \text{if } r \text{ is an edge from vertex } i \text{ to vertex } j, \\ 0, & \text{otherwise,} \end{cases} \quad ,$$

where $m = |E|$ is the total number of directed edges in the graph.

Example 2. **(Sampling by a Markov chain)** Given a directed graph $G = (V, E)$ with the adjacency matrix $A = (a_{i,j})$, consider an ergodic Markov chain on this graph. Let $p_{u,w}$ be the transition probability from node u to node w and π_u be the steady state probability of node u. In particular, for PageRank [6] with the web surfing probability λ, the transition probability of the corresponding Markov chain is

$$p_{u,w} = (1-\lambda)\frac{1}{n} + \lambda \frac{a_{u,w}}{k_u^{out}}, \tag{1}$$

where k_u^{out} is the out-degree of node u. Its steady state probabilities (with $\sum_{u=1}^n \pi_u = 1$) can be obtained from solving the following system of equations:

$$\pi_u = (1-\lambda)\frac{1}{n} + \lambda \sum_{w=1}^n \frac{a_{wu}}{k_w^{out}} \pi_w, \quad \text{for all } u = 1, 2, \ldots, n. \tag{2}$$

For a path r that traverses a sequence of nodes $\{u_1, u_2, \ldots, u_{k-1}, u_k\}$ in a directed network, we have from the Makrov property that

$$p(r) = \pi_{u_1} \cdot p_{u_1,u_2} \cdot \ldots \cdot p_{u_{k-1},u_k}. \tag{3}$$

Consider the reverse path of r, denoted by $Rev(r)$, that traverses a sequence of nodes $\{u_k, u_{k-1}, \ldots, u_2, u_1\}$. If the Markov chain is reversible (see e.g., the book [38]), then it follows from the detailed balance equation that

$$p(r) = p(Rev(r)). \tag{4}$$

It is known that the corresponding Markov chain for PageRank is in general not reversible. However, the corresponding Markov chain for a random walk on an undirected graph is reversible. We will discuss this further in the next example.

Example 3. **(Sampling by a random walk on an undirected network with path length 1 or 2)** For an undirected graph $G(V, E)$, let $m = |E|$ be the total number of edges and k_v be the degree of node v, $v = 1, 2, \ldots, n$. A path r with length 1 can be represented by the two nodes $\{u_1, u_2\}$ it traverses. Similarly, a path with length 2 can be represented by the three nodes $\{u_1, u_2, u_3\}$ it traverses. A random walk with path length not greater than 2 can be generated by the following two steps: (i) with the probability $k_v/2m$, an initial node v is chosen, (ii) with probability β_i, $i = 1, 2$, a walk with length i is chosen. As such, we have

$$p(r) = \begin{cases} \frac{\beta_1}{2m} a_{u_1,u_2}, & \text{if } r = \{u_1, u_2\}, \\ \frac{\beta_2}{2m} \frac{a_{u_1,u_2} a_{u_2,u_3}}{k_{u_2}}, & \text{if } r = \{u_1, u_2, u_3\}, \end{cases} \tag{5}$$

where $\beta_1 + \beta_2 = 1$ and $\beta_i \geq 0$, $i = 1, 2$. For an undirected network, we have $a_{i,j} = a_{j,i}$ for all $i, j = 1, 2, \ldots, n$. Thus, in view of (5), we also have

$$p(r) = p(Rev(r)), \tag{6}$$

where $Rev(r)$ is the reverse path of r. Moreover, if $\beta_2 = 0$, then the random walk has path length 1, and this is equivalent to sampling by uniformly selecting an edge.

Let U (resp. W) be the starting (resp. ending) node of a randomly selected path by using the path sampling distribution $p(\cdot)$. Then the bivariate distribution

$$p_{U,W}(u, w) = \mathsf{P}(U = u, W = w) = \sum_{r \in R_{u,w}} p(r) \tag{7}$$

is the probability that the ordered pair of two nodes (u, w) is selected. As such, $p_{U,W}(u, w)$ can be viewed as a similarity measure from node u to node w and this leads to the definition of a sampled graph in [9], [23].

Definition 4. **(Sampled graph [9], [23])** A graph $G(V, E)$ that is sampled by randomly selecting an ordered pair of two nodes (U, W) according to a specific bivariate distribution $p_{U,W}(\cdot, \cdot)$ in (7) is called a *sampled graph* and it is denoted by the two-tuple $(G(V, E), p_{U,W}(\cdot, \cdot))$.

Let $p_U(u)$ (resp. $p_W(w)$) be the marginal distribution of the random variable U (resp. W), i.e.,

$$p_U(u) = \mathsf{P}(U = u) = \sum_{w=1}^n p_{U,W}(u, w), \tag{8}$$

and

$$p_W(w) = \mathsf{P}(W = w) = \sum_{u=1}^n p_{U,W}(u, w). \tag{9}$$

Then $p_U(u)$ is the probability that node u is selected as a starting node of a path and it can be viewed as an out-centrality of u. On the other hand, $p_W(w)$ is the probability that node w is selected as an ending node of a path and it can be viewed as an in-centrality of w. For directed networks, e.g., citation networks [5], the in-centrality and the out-centrality of the degree centrality (sampling by uniformly selecting an edge in Example 1) is the in-degree (resp. out-degree) centrality that is represented by the number of incoming (resp. outgoing) edges. As commented in [5], the papers with high out-degree centralities in a citation network are usually survey papers that contain lots of references. On the other hand, highly-cited papers are the papers with high in-degree centralities. The in-centrality and the out-centrality are in general not the same. Clearly, if the bivariate distribution $p_{U,W}(\cdot, \cdot)$ is symmetric, then the in-centrality and the out-centrality are the same. A recent advance in

[23] shows that one does not need a symmetric bivariate distribution to ensure the equality between the in-centrality and the out-centrality. In particular, for the Markov chain sampling methods, one still has $p_U(u) = p_W(u)$ and the in-centrality and the out-centrality are the same. In that case, we will simply refer $P_U(u)$ as the centrality of node u.

Definition 5. **(Covariance, Community, and Modularity [9], [23])** For a sampled graph $(G(V,E), p_{U,W}(\cdot,\cdot))$, the covariance between two nodes u and w is defined as follows:

$$q(u,w) = p_{U,W}(u,w) - p_U(u)p_W(w). \tag{10}$$

Moreover, the covariance between two sets S_1 and S_2 is defined as follows:

$$q(S_1,S_2) = \sum_{u\in S_1}\sum_{w\in S_2} q(u,w). \tag{11}$$

Two sets S_1 and S_2 are said to be positively correlated if $q(S_1,S_2) \geq 0$. In particular, if a subset of nodes $S \subset V$ is positively correlated to itself, i.e., $q(S,S) \geq 0$, then it is called a *community* or a *cluster* (in this paper, we will use community and cluster interchangeably).

Let $\mathcal{P} = \{S_k, k = 1,2,\dots,K\}$, be a partition of V, i.e., $S_k \cap S_{k'}$ is an empty set for $k \neq k'$ and $\cup_{k=1}^K S_k = V$. The modularity $Q(\mathcal{P})$ with respect to the partition S_k, $k = 1,2,\dots,K$, is defined as

$$Q(\mathcal{P}) = \sum_{k=1}^K q(S_k,S_k). \tag{12}$$

There are many physical interpretations (and equivalent statements) for the definition of a community in [9], [23]. Moreover, as pointed out in the book [5], the physical meaning of the modularity with respect to a partition of a graph is how much it differs from that partition of a random graph generated by the configuration model. As such, a good partition of a graph should have a large modularity. In view of this, one can then tackle the community detection/clustering problem by looking for algorithms that yield large modularity. For this, we define the modularity matrix Γ for the sampled graph $(G(V,E), p_{U,W}(\cdot,\cdot))$ as the $n \times n$ matrix with its $(u,w)^{th}$ element being $q(u,w)$ in (10). Then the modularity maximization problem can be formulated as the optimization problem that finds an $n \times K$ partition matrix H_K to maximize $tr(H_K^T \Gamma H_K)$ over K and H_K. The modularity maximization problem is known to be NP-hard [39] and one has to resort to heuristic algorithms. In the literature, there are several community detection algorithms that find a partition to achieve a local maximum of the modularity in (12), e.g., the spectral modularity maximization algorithm [40], the hierarchical agglomerative algorithm [15], the partitional algorithm [41], and (iv) the fast unfolding algorithm [42]. For a detailed introduction of these algorithms, we refer to [23].

3 Exponentially twisted sampling in attributed networks

Now we generalize the probabilistic framework in [9], [23] to attributed networks. In order to take the attributes of nodes and edges into account, the sampling method in an attributed network and the corresponding probability measure needs to be a twisted probability measure of its underlining network. This then leads to the exponentially twisted probability measure [32], [33], [34].

An attributed network is a generalization of a graph $G(V,E)$ by assigning each node $u \in V$ an attribute $h_V(u)$ and each edge $e \in E$ an attribute $h_E(e)$. As such, an attributed network can be represented as $G(V,E,h_V(\cdot),h_E(\cdot))$, where $h_V(\cdot)$ and $h_E(\cdot)$ are called the node attribute function and the edge attribute function, respectively.

For a path r that traverses a sequence of nodes $\{u_1,u_2,\dots,u_{k-1},u_k\}$ in an attributed network, we can define a path measure $f(r)$ as a function of the attributes of the nodes and edges along the path r, i.e.,

$$\{h_V(u_1),\dots,h_V(u_k),h_E(u_1,u_2),\dots,h_E(u_{k-1},u_k)\}.$$

In this paper, we assume that a path measure $f(\cdot)$ is a mapping from the set of paths R to an L-dimensional real-valued vector in $\mathcal{R}^L$, i.e.,

$$f(r) = (f_1(r), f_2(r), \dots, f_L(r)),$$

where $f_i(\cdot)$, $i = 1,2,\dots,L$, are real-valued functions.

Now suppose that we have already had a sampled graph $(G(V,E), p_0(\cdot))$ that uses the probability mass function $p_0(r)$ to sample a path r in $G(V,E)$. The question is how the sampling distribution should be changed so that the average path measure is equal to a specified vector $\overline{f}$. In other words, what is the most likely sampling distribution $p(\cdot)$ that leads to the average path measure $\overline{f}$ given that the original sampling distribution is $p_0(\cdot)$? For this, we introduce the Kullback-Leibler distance between two probability mass functions $p(\cdot)$ and $p_0(\cdot)$:

$$D(p\|p_0) = \sum_{r\in R} p(r)\log(\frac{p(r)}{p_0(r)}). \tag{13}$$

The Kullback-Leibler distance is known to be nonnegative and it is zero if and only if $p(\cdot) = p_0(\cdot)$ (see e.g., [33]). Also, according to the Sanov theorem (see e.g., the books [33], [34]), the larger the Kullback-Leibler distance is, the more unlikely for $p_0(\cdot)$ to behave like $p(\cdot)$. Thus, to address the question, we consider the following constrained minimization problem:

$$\begin{aligned} \min \quad & D(p\|p_0) \\ s.t. \quad & \sum_{r\in R} p(r) = 1, \\ & \sum_{r\in R} f(r)p(r) = \overline{f}. \end{aligned} \tag{14}$$

The first constraint states that the total probability must be equal to 1. The second constraint states that the average path measure must be equal to $\overline{f}$ with the new sampling distribution $p(\cdot)$.

The above minimization problem can be solved by using Lagrange's multipliers $\alpha \in \mathcal{R}$ and $\theta \in \mathcal{R}^L$ as follows:

$$\begin{aligned} I = D(p\|p_0) + \alpha(1 - \sum_{r\in R} p(r)) \\ +\theta \cdot (\overline{f} - \sum_{r\in R} f(r)p(r)). \end{aligned} \tag{15}$$

Taking the partial derivative with respect to $p(r)$ yields

$$\frac{\partial I}{\partial p(r)} = \log p(r) + 1 - \log p_0(r) - \alpha - \theta \cdot f(r) = 0. \quad (16)$$

Thus,

$$p(r) = \exp(\alpha - 1) * \exp(\theta \cdot f(r)) * p_0(r). \quad (17)$$

Since $\sum_{r \in R} p(r) = 1$, it then follows that

$$p(r) = C * \exp(\theta \cdot f(r)) * p_0(r), \quad (18)$$

where

$$C = \frac{1}{\sum_{r \in R} \exp(\theta \cdot f(r)) * p_0(r)} \quad (19)$$

is the normalization constant.

To solve the parameter vector θ, we let

$$F = \log(1/C).$$

The quantity F is called the free energy as it is analogous to the free energy in statistical mechanic [5]. Also, the parameter vector θ is called the inverse temperature vector. It is easy to see that for $i = 1, 2, \ldots, L$ that

$$\frac{\partial F}{\partial \theta_i} = \sum_{r \in R} f_i(r) p(r) = \bar{f}_i. \quad (20)$$

These L equations can then be used to solve θ_i, $i = 1, 2, \ldots, L$.

Once we have the sampling distribution in (18), we can define a bivariate distribution $p_{U,W}(u, w)$ as in (7). Analogous to the discussion of a sampled graph in the previous section, the marginal distribution of the random variable U (resp. W), i.e., $p_U(u)$ (resp. $p_W(w)$), can be viewed as an out-centrality of u (resp. in-centrality of w).

To summarize, in order to define the out-centrality and the in-centrality of an attributed network, one needs (i) the original sampling distribution $p_0(\cdot)$ for the network, and (ii) the path measure $f(\cdot)$ of the attributed network. Once a specified average path measure $\bar{f}$ (or the inverse temperature vector θ) is given, one can have the new sampling distribution $p(\cdot)$ in (18). This then leads to the bivariate distribution in (7). The marginal distributions of that bivariate distribution then correspond to the out-centrality and the in-centrality of the attributed network.

4 Path Measures in Attributed Networks

In this section, we introduce three path measures in attributed networks and these lead to influence centralities in signed networks in Section 4.1, the trust centralities in signed networks in Section 4.2, and the advertisement-specific influence centralities in networks with node attributes in Section 4.3. For community detection, we consider an attributed network with a distance measure in Section 4.4. We show communities with different sizes can be detected by specifying a resolution parameter derived from the average distance between two randomly sampled nodes.

4.1 Influence centralities in signed networks

In this section, we consider a special class of attributed networks, called *signed networks*. A signed network $G = (V, E, h_E(\cdot))$ is an attributed network with an edge attribute function $h_E(\cdot)$ that maps every undirected edge in E to the two signs $\{+, -\}$. In this paper, we represent the positive (resp. negative) sign by 1 (resp. -1). An edge (u, w) mapped with the $+$ sign is called a *positive* edge, and it is generally used for indicating the *friendship* between the two nodes u and w. On the other hand, an edge mapped with the $-$ sign is called a *negative* edge. A negative edge (u, w) indicates that u and w are *enemies*.

One interesting question for signed networks is how the nodes in signed networks are ranked. Our idea for this is to use opinion dynamics. If u and w are connected by a positive (resp. negative) edge, then it is very likely that u will have a positive (resp. negative) influence on w and vice versa. As such, if we start from a node u with a positive opinion on a certain topic, then a neighbor of node u connected by a positive (resp. negative) edge will tend to have the same (resp. the opposite) opinion as node u has. Now we can let the opinion propagate through the entire network (via a certain opinion dynamic) and count the (expected) number of nodes that have the same opinion as node u has. If such a number is large, then it seems reasonable to say that node u has a large positive influence on the other nodes in the network. In other words, a node u has a large positive influence if there is a high probability that the other end of a randomly selected path has the same opinion as node u. This then leads us to define the notion of *influence centralities* for ranking nodes in signed networks.

The above argument is based on the general belief that "a friend of my friend is likely to be my friend" and "an enemy of my enemy can be my friend" in [5]. As such, for a path r that traverses a sequence of nodes $\{u_1, u_2, \ldots, u_k\}$ in a signed network, we define the following path measure as the product of the edge signs along the path, i.e.,

$$f(r) = \prod_{(u_i, u_{i+1}) \in r} h_E(u_i, u_{i+1}). \quad (21)$$

Note that $f(r)$ is either 1 or -1 as the edge attribute function $h_E(\cdot)$ that maps every undirected edge in E to $\{1, -1\}$.

As an illustrating example, let us consider using the sampling distribution $p_0(r)$ by a random walk with path length 1 or 2 in Example 3. It then follows from (18), (21) and (5) that

$$p(r) = \begin{cases} C \cdot e^{\theta h_E(u_1,u_2)} \cdot \frac{\beta_1}{2m} a_{u_1,u_2}, \\ \quad \text{if } r = \{u_1, u_2\}, \\ C \cdot e^{\theta \cdot h_E(u_1,u_2) \cdot h_E(u_2,u_3)} \cdot \frac{\beta_2}{2m} \frac{a_{u_1,u_2} a_{u_2,u_3}}{k_{u_2}}, \\ \quad \text{if } r = \{u_1, u_2, u_3\}. \end{cases} \quad (22)$$

The constant C in (22) is the normalization constant. Summing all the paths from u to w yields the bivariate distribution

$$p_{U,W}(u, w) = C\Big[e^{\theta h_E(u,w)} \cdot \frac{\beta_1}{2m} a_{u,w} + \sum_{u_2 \in V} e^{\theta \cdot h_E(u,u_2) \cdot h_E(u_2,w)} \cdot \frac{\beta_2}{2m} \frac{a_{u,u_2} a_{u_2,w}}{k_{u_2}}\Big]. \quad (23)$$

The marginal distribution of the bivariate distribution, denoted by $P_U(u)$, is called the *influence centrality* of node u (with respect to the inverse temperature θ).

If we only select paths with length 1, i.e., $\beta_2 = 0$ in (23), then there is a closed-form expression for the influence centrality. For this, we first compute the normalization constant C by summing over u and w in (23) and this yields

$$C = \frac{m}{m^+ e^\theta + m^- e^{-\theta}}, \tag{24}$$

where m^+ (resp. m^-) is the total number of positive (resp. negative) edges in the graph. Thus, for $\beta_2 = 0$,

$$\begin{aligned} p_U(u) &= \sum_{w \in V} p_{U,W}(u,w) \\ &= \frac{(k_u^+ e^\theta + k_u^- e^{-\theta})}{2(m^+ e^\theta + m^- e^{-\theta})}, \end{aligned} \tag{25}$$

where k_u^+ (resp. k_u^-) is the number of positive (resp. negative) edges of node u.

Now suppose we require the average path measure $\bar{f}$ to be equal to some fixed constant $-1 < \gamma < 1$. Then we have from (20) that

$$\begin{aligned} \gamma &= \bar{f} = \frac{\partial F}{\partial \theta} \\ &= \frac{m^+ \exp(\theta) - m^- \exp(-\theta)}{m^+ \exp(\theta) + m^- \exp(-\theta)}, \end{aligned} \tag{26}$$

where $F = \log(1/C)$ with C in (24) being the free energy. This then leads to

$$\theta = \ln(\sqrt[2]{\frac{m^-(1+\gamma)}{m^+(1-\gamma)}}). \tag{27}$$

Now we discuss the connection of the influence centralities with the three degree ranking methods: (i) ranking by the number of positive edges (positive degree), (ii) ranking by the number of negative edges (negative degree), and (iii) ranking by the total number of edges (total degree). When $\gamma \to 1$, we have from (27) that $\theta \to \infty$. As a result from (25), $P_U(u) \to \frac{k_u^+}{2m^+}$ and this corresponds to positive degree ranking. On the other hand, when $\gamma \to -1$, we have $\theta \to -\infty$ and $P_U(u) \to \frac{k_u^-}{2m^-}$. This corresponds to negative degree ranking. Finally, if we choose $\gamma = (m^+ - m^-)/(m^+ + m^-)$, then $\theta = 0$ and $P_U(u) = \frac{k_u}{2m}$. This corresponds to total degree ranking. Thus, different choices of γ lead to different ranking methods. We will illustrate this further in Section 5.1.

4.2 Trust centralities in signed networks

As discussed in the previous section, the influence centralities are based on the general belief that "an enemy of my enemy can be my friend." Such a statement might be valid for modelling opinion dynamics. However, it is not suitable for modelling *trust*. In addition to the interpretation of a signed edge as the friend/enemy relationship, another commonly used interpretation is the trusted/untrusted link. A path r that traverses a sequence of nodes $\{u_1, u_2, \ldots, u_k\}$ can be *trusted* if every edge is a trusted link so that there exists a chain of trust. In view of this, the notion of trust centrality in a signed network can be defined by using the path measure f that is the minimum of the edge signs along the path, i.e.,

$$f(r) = \min_{(u_i, u_{i+1}) \in r} h(u_i, u_{i+1}). \tag{28}$$

4.3 Advertisement-specific influence centralities in networks with node attributes

In this section, we consider another class of attributed networks that have node attributes. For a graph $G(V, E)$ with the node attribute function $h_V(u)$ that maps every node u to a vector in $\mathcal{R}^L$

$$(h_{V,1}(u), h_{V,2}(u), \ldots, h_{V,L}(u)). \tag{29}$$

One intuitive way to interpret such an attributed network is to view the graph $G(V, E)$ as a social network with n users and the attribute vector in (29) as the scores of user u on various topics. Now suppose an advertisement z can be represented by a vector of scores $(z_1, z_2, \ldots, z_L)$ with z_i being the score of the i^{th} topic. Then we would like to find out who is the most influential user in the network to pass on the advertisement z. Such a problem was previously studied in [26] for ranking nodes in Twitter. In TwitterRank [26], a two-step approach was used. First, a topic-specific ranking is obtained for each topic by using a random surfer model similar to that in PageRank [6]. The second step is then to take the weighted average over these topic-specific rankings. Specifically, suppose that $RT_i(u)$ is the ranking for topic i and user u. TwitterRank for advertisement z and user u is then defined as the following weighted average:

$$\sum_{i=1}^{L} z_i \cdot RT_i(u). \tag{30}$$

One flaw for such a two-step approach is that it neglects the fact that the propagation of a specific advertisement through a user depends on how much a user "likes" the advertisement. To model how much a user "likes" an advertisement, we use the similarity measure from the inner product of the score vector of the user and that of the advertisement. It is possible that in a cascade of two users $\{u_1, u_2\}$, both users like the advertisement because their inner products are large, but user u_1 likes one topic in that advertisement and user u_2 likes another different topic in that advertisement. Such a cascade cannot be modelled by using the two-step approach in TwitterRank [26]. In view of this, it might be better to use a one-step approach for computing advertisement-specific influence centralities. As the influence centralities in the previous section, we propose using opinion dynamics through a path. For a path r that traverses a sequence of nodes $\{u_1, u_2, \ldots, u_{k-1}, u_k\}$ in the attributed network, we define the following path measure

$$f(r) = \min_{u \in r}[\sum_{i=1}^{L} z_i \cdot h_{V,i}(u)]. \tag{31}$$

4.4 Clustering with a distance measure

In this section, we consider another class of attributed networks in which the edge attribute of a directed edge is the distance from one end to the other end of the directed edge. Denote such a network by $G = (V, E, d(\cdot, \cdot))$, where $d(u, w)$

is the distance from node u to node w. Let $n = |V|$ be the number of nodes in the graph. In this paper, we assume that the distance is nonnegative, i.e., $d(u, w) \geq 0$. Also, we can extend the definition of the distance between two nodes that are not connected by a directed edge by setting the distance to be infinity, i.e., $d(u, w) = \infty$ if there does not exist a directed edge from node u to node w. We also add n self edges by letting $d(u, u) = 0$ for all the n nodes. By doing so, we have a complete graph with n self edges and the total number of directed edges m is n^2.

A natural selection of a path measure in such a complete graph with n self edges is the total distance along a (directed) path, i.e.,

$$f(r) = \sum_{(u_i,u_{i+1})\in r} d(u_i, u_{i+1}), \tag{32}$$

for a path r that traverses a sequence of nodes $\{u_1, u_2, \ldots, u_k\}$ in such a complete graph with n self edges.

As an illustrating example, let us consider using the sampling distribution $p_0(r)$ by uniformly selecting a directed edge in Example 1, i.e.,

$$p_0(r) = \begin{cases} 1/n^2, & \text{if } r = \{u_1, u_2\}, \\ 0, & \text{otherwise.} \end{cases} \tag{33}$$

From (18) and (33), we have the following exponentially twisted sampling distribution

$$p(r) = \begin{cases} C \cdot \exp(\theta \cdot d(u_1, u_2)) \cdot \frac{1}{n^2}, & \text{if } r = \{u_1, u_2\}, \\ 0, & \text{otherwise,} \end{cases} \tag{34}$$

where C in (34) is the normalization constant. Summing up all the n^2 directed edges yields

$$C = \frac{n^2}{\sum_{u_1\in V}\sum_{u_2\in V}\exp(\theta \cdot d(u_1, u_2))}.$$

This then leads to the following bivariate distribution

$$p_{U,W}(u, w) = \frac{\exp(\theta \cdot d(u, w))}{\sum_{u_1}\sum_{u_2}\exp(\theta \cdot d(u_1, u_2))}. \tag{35}$$

Note that under the original uniform sampling distribution, the average distance between two randomly selected nodes U and W is

$$\mathsf{E}_{p_0}[d(U, W)] = \frac{1}{n^2}\sum_{u\in V}\sum_{w\in V} d(u, w). \tag{36}$$

On the other hand, the average distance between two randomly selected nodes U and W under the exponentially twisted sampling distribution is

$$\mathsf{E}_p[d(U, W)] = \frac{\sum_{u\in V}\sum_{w\in V} d(u, w)\exp(\theta \cdot d(u, w))}{\sum_{u_1}\sum_{u_2}\exp(\theta \cdot d(u_1, u_2))}. \tag{37}$$

To solve the parameter θ, we can specify the average distance between two randomly selected nodes under the exponentially twisted sampling distribution to be a desired value $\bar{d}$, i.e.,

$$\mathsf{E}_p[d(U, W)] = \bar{d}. \tag{38}$$

For the clustering purpose, in general one should choose $\bar{d}$ to be smaller than $\mathsf{E}_{p_0}[d(U, W)]$ in (36) so that a pair of two nodes with a shorter distance is selected more often than another pair of two nodes with a larger distance. Clearly, if we choose $\theta < 0$, then $\bar{d} \leq \mathsf{E}_{p_0}[d(U, W)]$. Also, as $\theta \to \infty$, $\bar{d}$ approaches to the maximum distance between a pair of two nodes. On the other hand, as $\theta \to -\infty$, $\bar{d}$ approaches to the minimum distance between a pair of two nodes.

Once we have the bivariate distribution in (35), we can perform community detection (or clustering) by using modularity maximization algorithms as discussed in Section 2. The parameter θ serves as a resolution parameter that can be used for detecting communities with different sizes (we will illustrate this further in Section 5.4). In particular, for clustering data points in a Euclidean space, one may simply choose the distance measure between two points (nodes) as the square of the Euclidean distance, i.e., for two data points x and y in a Euclidean space, $d(x, y) = \|x - y\|^2$. Then the exponentially twisted sampling associated with a particular resolution parameter θ can be viewed as a transformation for data points in a Euclidean space to a sampled graph. Such a transformation is related to the transformation used in [43] (for spectral clustering) and [44] (for support vector clustering).

For our experiments in Section 5.4, we will consider distance measures that are *semi-metrics*. A distance measure $d(u, w)$ is called a semi-metric if it satisfies the following three properties:

(D1) (Nonnegativity) $d(u, w) \geq 0$.
(D2) (Null condition) $d(u, u) = 0$.
(D3) (Symmetry) $d(u, w) = d(w, u)$.

It is called a *metric* if the distance measure also satisfies the triangular inequality, i.e.,

(D4) (Triangular inequality) $d(u, w) \leq d(u, v) + d(v, w)$.

For the bivariate distribution in (35), let us consider the covariance measure $q(u, w)$ in (10) when θ is very small. Using the first order approximation $e^{\theta z} \approx 1 + \theta z + o(\theta)$ in (10) yields

$$\begin{aligned} q(u, w) \quad &\approx \quad (-\theta)\Big(\frac{1}{n}\sum_{u_2\in V} d(u_2, w) + \frac{1}{n}\sum_{u_1\in V} d(u, u_1) \\ &\qquad -\frac{1}{n^2}\sum_{u_2\in V}\sum_{u_1\in V} d(u_2, u_1) - d(u, w)\Big) + o(\theta). \end{aligned}$$

Thus, when we choose a very small negative θ, the covariance measure $q(u, w)$ is proportional to

$$\begin{aligned} \gamma(u, w) \quad &= \quad \frac{1}{n^2}\sum_{u_2\in V}\sum_{u_1\in V}\Big(d(u, u_1) + d(u_2, w) \\ &\qquad -d(u_1, u_2) - d(u, w)\Big). \end{aligned} \tag{39}$$

The function $\gamma(u, w)$ is called a cohesion measure (resp. semi-cohesion measure) if the distance measure $d(x, y)$ is a metric (resp. semi-metric) in [41], [45]. Analogous to (11), one can define the cohesion measure between two sets S_1 and S_2 as follows:

$$\gamma(S_1, S_2) = \sum_{u\in S_1}\sum_{w\in S_2}\gamma(u, w). \tag{40}$$

Since the covariance measure $q(u, w)$ is proportional to the cohesion measure $\gamma(u, w)$ for a small negative θ, community detection can be done directly by finding a partition $\{S_k, k = 1, 2, \ldots, K\}$ that maximizes $\sum_{k=1}^{K}\gamma(S_k, S_k)$. Such a maximization problem is known as the modularity maximization problem and it was shown to be NP-hard [39]. There are various heuristic algorithms proposed in the literature, including the spectral modularity maximization algorithm [40], the hierarchical agglomerative algorithm [15], the partitional algorithm [41], and the fast unfolding algorithm [42]. Among these algorithms, the partitional algorithm is a linear-time algorithm with the computation complexity $O((n+m)I)$, where I is the number of iterations [23].

5 Experiments

In this section, we conduct several experiments to evaluate our framework for centrality analysis and community detection in attributed networks. Here we report our results for the influence centralities in signed networks in Section 5.1, the trust centralities in Section 5.2, the advertisement-specific influence centralities in Section 5.3 and clustering with a distance measure in Section 5.4. Additional experimental results can be found in the full report [46].

5.1 Experimental results for the influence centralities in signed networks

In this section, we evaluate the influence centralities in signed networks by using the real dataset from the political blogs in [35]. The network in [35] is a directed network of hyperlinks between weblogs collected around the time of the United States presidential election of 2004. There are 1,490 nodes and 19,090 edges in this dataset. These 1490 nodes can be partitioned into two (ground-truth) communities (parties). In order to have a signed network for our experiment, we add 4,000 negative edges between two nodes chosen randomly from each community. We then symmetrize the adjacency matrix by using the matrix $A + A^T$ (to obtain an undirected network). We also delete the nodes with degree smaller than 7, and remove self edges and multiple edges. As a result, we obtain a simple undirected signed network with 863 nodes and 16,650 edges, including 15,225 positive edges and 1,425 negative edges.

For our experiment, we use sampling by a random walk with path length 1, i.e., $\beta_2 = 0$ in Example 3. In Figure 1, we show the ranking results of the influence centralities for six values of γ. The corresponding θ is computed from (27). The top 100 nodes are marked with different colors, 1-20 in red, 21-40 in orange, 41-60 in yellow, 61-80 in green and 81-100 in blue. When γ is chosen to be -0.9 in Figure 1(a), the top 100 nodes seem to be uniformly distributed among all the nodes. It is interesting to see that the top 100 nodes gradually move toward the "center" of each community when the value of γ is increased. Also, these top 100 nodes are separated into the two communities, and they are closely packed with each other around the center of each community. In our plots, the nodes near the center of each community have large degrees. As discussed in Section 4.1, the choice of γ is closely related to the three degree ranking methods: (i) ranking by the number of positive edges (positive degree), (ii) ranking by the number of negative edges (negative degree), and (iii) ranking by the total number of edges (total degree). To further illustrate the connection between the influence centrality and the three degree ranking methods, we compute the Jaccard index between the top 100 nodes obtained from the influence centrality and the top 100 nodes obtained from each of the degree centrality method. Recall that the Jaccard index between two sets S_1 and S_2 is computed as follows:

$$J(S_1, S_2) = \frac{|S_1 \cap S_2|}{|S_1 \cup S_2|}. \tag{41}$$

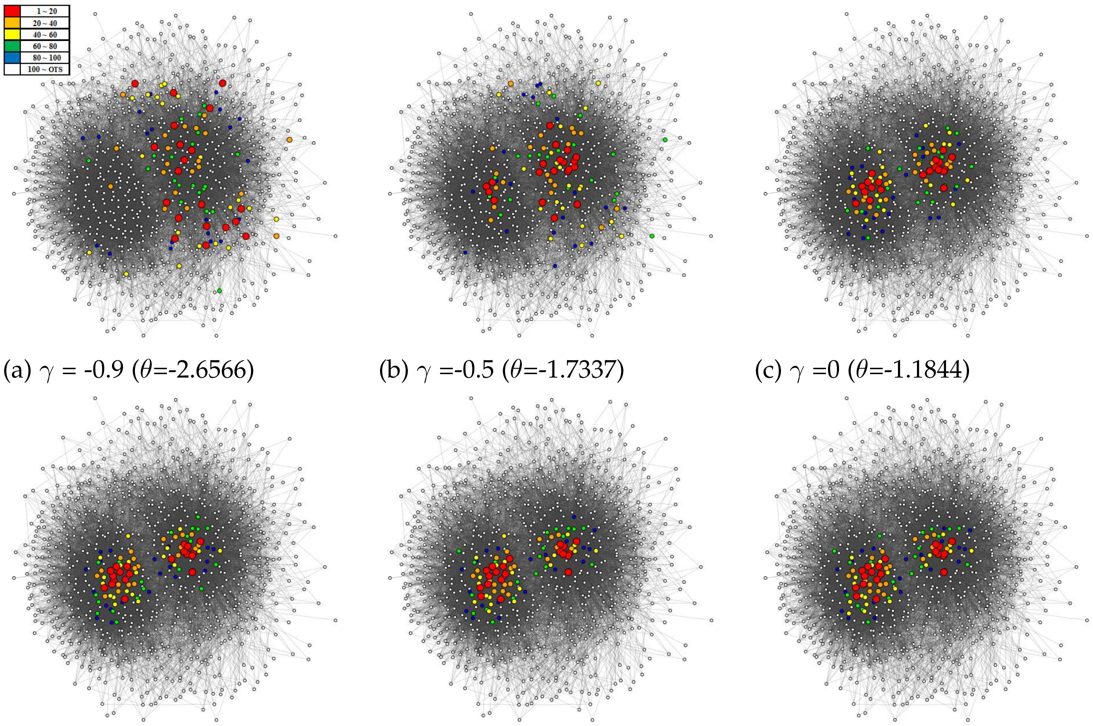
In Figure 2, we plot the three curves of the Jaccard indices with respect to γ. One can see that the Jaccard index for the curve from ranking by negative degree is a decreasing function of γ, while the other two curves are increasing functions of γ. This shows that the influence centrality with γ close to -1 is mostly in line with ranking by negative degree. Intuitively, one can view nodes with high negative degrees as *speakers* of a party who have the tendency to criticize the other party. On the other hand, the influence centrality with γ close to 1 is mostly in line with ranking by positive degree. This is because increasing γ increases the probability that a positive edge is sampled (and decreases the probability that a negative edge is sampled). Nodes with high positive degrees can be viewed as *party leaders* of a party. The party leaders of these two parties (clusters) can be easily found as shown in Figure 1(f). Note that there is a slight difference between ranking by positive degree and ranking by total degree when γ is close to 1 as ranking by total degree still counts the number of negative edges and those negative edges have little chance being selected when γ is close to 1.

5.2 Experimental results for the trust centralities in signed networks

We use the same signed network as that in the previous section. For the trust centrality, we consider sampling by a random walk with path length 1 or 2, where $\beta_1 = 0.7$ and $\beta_2 = 0.3$ in Example 3. Note that if we use sampling by a random walk with path length 1, then the trust centrality obtained this way is the same as the influence centrality. As shown in Figure 3, the ranking results do not change very much for various values of θ. For a path with length 2, its path measure is 1 only when the two edges traversed by the path are positive edges. As such, nodes that have a large number of positive edges tend to have high trust centralities (for a wide range of θ). For this dataset, 91.5% of edges are positive edges. Thus, when θ is larger than zero, the ranking result is similar to that by the positive degree and that by the total degree.

5.3 Experimental results for the advertisement-specific influence centralities

In this section, we evaluate the performance of the advertisement-specific influence centrality by using the MemeTracker dataset [36]. Such a dataset collects the quotes and phrases, called "memes," that frequently appear over time across mass media and personal blogs. To obtain the advertisement information from these memes, we use Carrot2 [47], an open source clustering engine, to classify memes into fifteen topics including "People", "Going", "Know", "Years", "Way", "United States", "States", "Life", "Believe", "Lot", "Love", "America", "Country", "Barack Obama" and "Obama". As "United States" and "Barack



(a) γ = -0.9 (θ=-2.6566)

(b) γ =-0.5 (θ=-1.7337)

(c) γ =0 (θ=-1.1844)

(d) γ =0.5 (θ=-0.6351)

(e) γ =0.9 (θ=0.2878)

(f) γ =0.99 (θ=1.4623)

Fig. 1. The ranking results of the influence centrality by sampling with β_2=0 for various values of θ

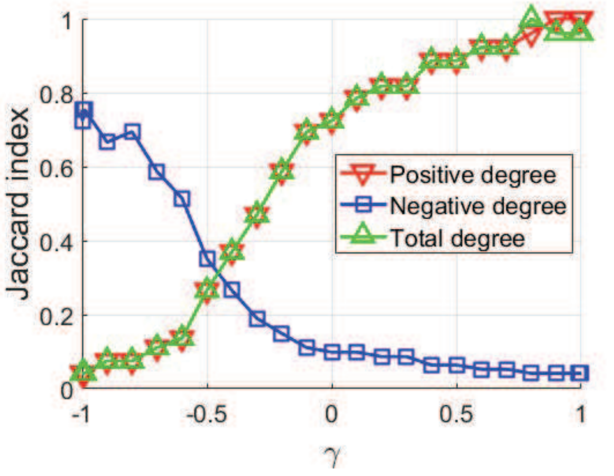


Fig. 2. The three Jaccard index curves between the influence centralities and the three degree centralities for the top 100 nodes.

Obama" are clearly subsets of the two topics, "States" and "Obama", they are merged into these two topics. Therefore, we obtain a dataset with thirteen topics. As in the previous experiment, we delete the nodes with degree smaller than 7, and remove self edges and multiple edges. This then leads to an attributed network with 2082 nodes and 16,503 edges.

Again, we use sampling by a random walk with path length 1, i.e., $\beta_2 = 0$ in Example 3. The inverse temperature θ is set to be 0.2 (as the top 250 nodes do not change when θ is larger than 0.2 in our experiments). In Figure 4, we show the ranking results for the six most frequently-used phrases, i.e., "Going", "Know", "People", "Years", "America", and "Obama". As shown in Figure 4, different topics lead to different ranking results.

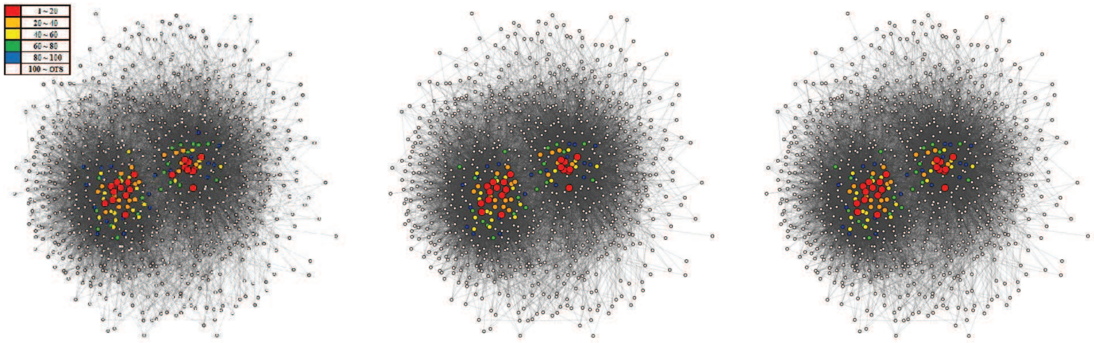
In addition, we also perform the ranking experiments by combining various topics. In Figure 5(a), we show the ranking result of advertisement-specific influence centralities by combining the two topics "Going" and "Obama." In Figure 5(b), we show the ranking result of advertisement-specific influence centralities by combining the two topics "Know" and "America." These ranking results are not necessarily the same as that from each topic.

5.4 Experimental results for clustering with a distance measure

5.4.1 The effect of the resolution parameter

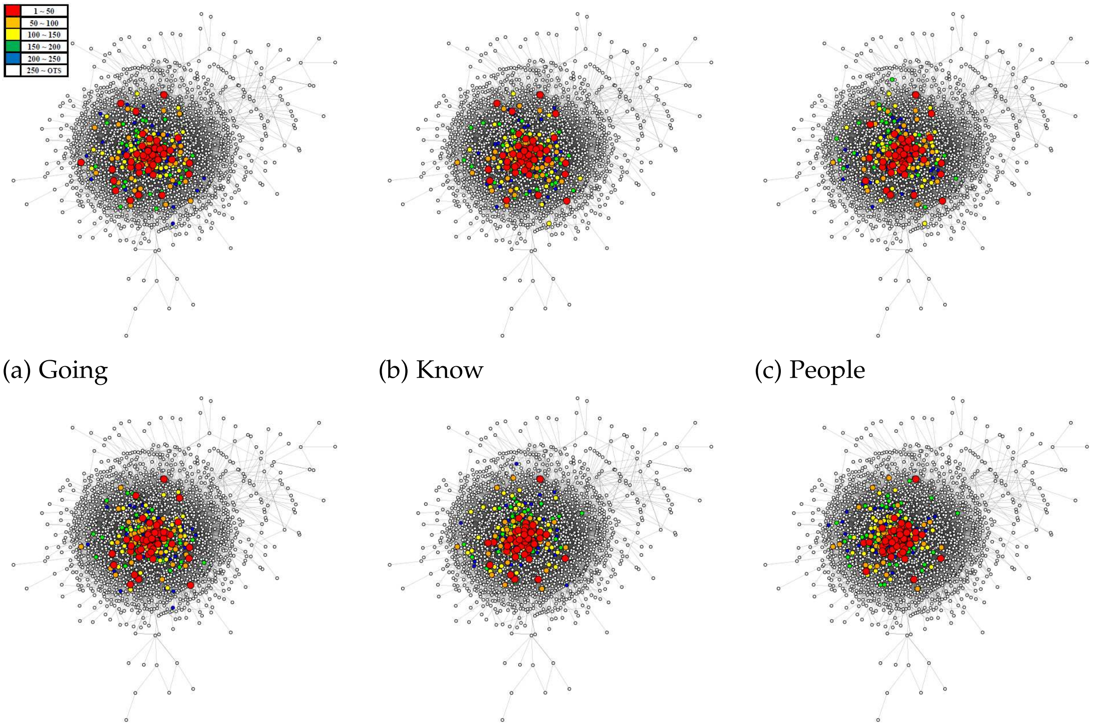
In this section, we illustrate how the parameter θ in the exponential twisted sampling in (35) can be used as a resolution parameter for detecting clusters with different sizes. In Fig. 6, we plot a set of 1250 data points on a plane with different scales of the two axes. A quick glance at this figure might yield five (resp. four, three and one) clusters in Fig. 6 (a) (resp. (b), (c) and (d)). As such, the problem of clustering is in general considered as an ill-posed problem [22], [48] and the answer of the number of clusters in the same dataset usually depends on how one "views" this dataset.

To see how the parameter θ in the exponential twisted sampling in (35) can be used as a resolution parameter for detecting clusters with different sizes for this data set, we use the distance measure that is the square of the Euclidean



(a) θ =0 (b) θ =1 (c) θ =2

Fig. 3. The ranking results of the trust centrality by sampling with β_1=0.7 and β_2=0.3 for various values of θ



(a) Going (b) Know (c) People

(a) Years (b) America (c) Obama

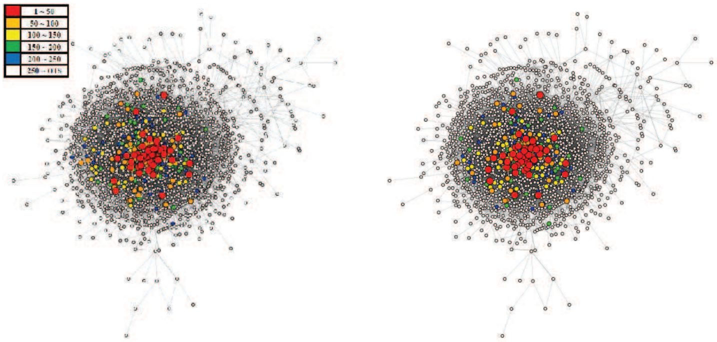
Fig. 4. The ranking results of advertisement-specific influence centralities with different topics

distance between two points. In Figure 7, we plot the average distance $\bar{d}$ under the exponentially twisting sampling as a function of θ. The plot in Fig. 7 allows us to solve θ in (37) and (38) numerically for each given $\bar{d}$.

As shown in Fig. 7, one can select a particular θ for the exponential twisted sampling distribution so that the average distance is $\bar{d}$. Once θ is determined, we have the bivariate distribution in (35) and we then perform community detection (or clustering) by using a modularity maximization algorithm. For our experiment, we use the partitional-hierarchical algorithm [23] that consists of two phases: the partition algorithm in the first phase and the hierarchical agglomerative algorithm in the second phase (that takes the output from the first phase as its input). In Table 1, we list the average distance $\bar{d}$ for various choices of θ. It is clear to see from Fig. 8 that various choices of θ lead to various resolutions of the clustering algorithm. Specifically, for $\theta = -0.5$ (and $\bar{d} = 2.6055$), there are five clusters detected by the partitional-hierarchical algorithm. Data points in different clusters are marked with different colors. For $\theta = -0.01$ (and $\bar{d} = 31.1958$), there are four clusters detected by the partitional-hierarchical algorithm. Finally, for $\theta = -0.0001$ (and $\bar{d} = 36.6545$), there are only three clusters detected by the partitional-hierarchical algorithm. These clustering results obtained by using different sampling distributions are in line with those plotted by using different scales of the two axes in Fig. 6.

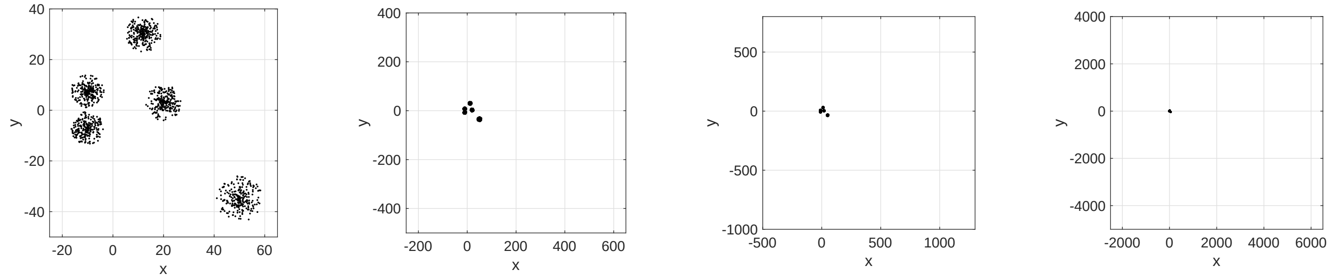
TABLE 1
The average distance $\bar{d}$ for various choices of θ.

θ	-0.5	-0.01	-0.0001	0	1
$\bar{d}$	2.6055	31.1958	36.6545	36.7121	87.8800



(a) Going and Obama (b) Know and America

Fig. 5. The ranking results of advertisement-specific influence centralities from combining two topics



(a) Scale 1 (b) Scale 2 (c) Scale 3 (d) Scale 4

Fig. 6. A dataset plotted with different scales of the two axes.

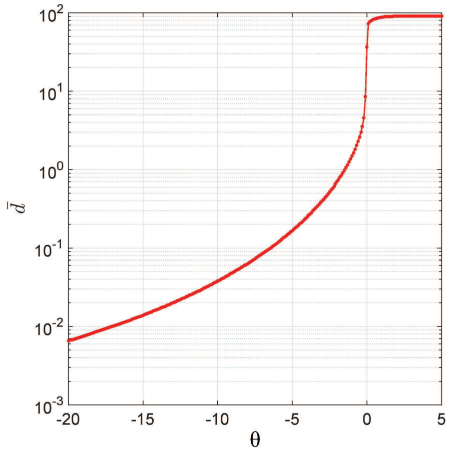
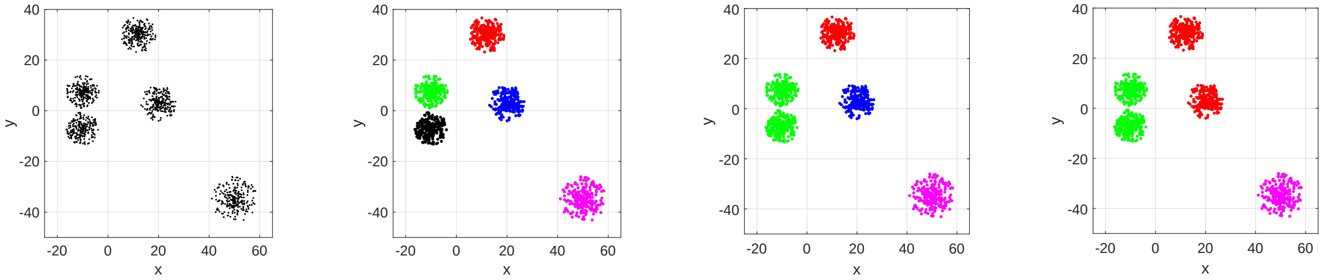


Fig. 7. The average distance $\bar{d}$ as a function of θ for the dataset in Fig. 6(a).

5.4.2 Clustering of a real network

In this section, we consider the real network from the WonderNetwork website [37]. Such a dataset was previously used in [45] to test the performance of the K-set$^+$ clustering algorithm. In this dataset, there are 216 servers in different locations and the latency (measured by the round trip time) between any two servers of these 216 servers are recorded in real time. As in [45], we symmetrize the latency matrix by taking the average of the latency measures from both directions so that the latency measure is a semi-metric. In addition to the latency measure, we also compute the distance measure by using the geographic location of each server in the WonderNetwork website. For such a dataset, we compute the corresponding semi-cohesion measures in (39) from the latency measure and the distance measure, respectively. We then use the corresponding semi-cohesion measure as the input of the partitional-hierarchical algorithm and run 20 times of the algorithm. In each of the 20 trials, the initial partition is randomly selected. The output partition that has the best objective value from these 20 trials is selected. The results for the distance measure and the latency measure are shown in Figure 9(a) and (b), respectively. For the K-set$^+$ clustering algorithm, one needs to specify the number of clusters K and K was to set to be 5 in the experiments in [45]. On the other hand, there is no need to specify the number of clusters K in the partitional-hierarchical algorithm. As shown in Figure 9(a) and (b), the partitional-hierarchical algorithm outputs three clusters (with the servers in the same cluster being marked with the same colored marker). In Figure 9(a), the three clusters are (i) North America and South America (marked in red), (ii) Europe (marked in green), and (iii) Asia, Africa and Australia (marked in red). Note that the key difference between Figure 9(a) and (b) is that the servers in Africa are clustered with the servers in Europe for the latency measure. This is because there are a few connected cables between Africa and Europe (as discussed in [45] by using the Submarine Cable Map [49]). Also, there are four servers, two in Russia (Vladivostok and Novosibirsk), one in Karaganda and one in Lahore, that have low latency to the servers in Europe and they are clustered with the servers in Europe. Even though New Delhi and Lahore are geographically close, however,



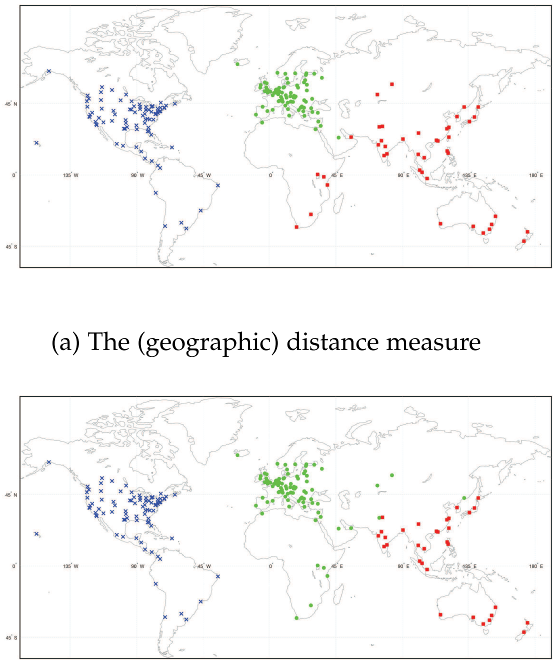
(a) The original dataset

(b) $\theta = -0.5$

(c) $\theta = -0.01$

(d) $\theta = -0.0001$

Fig. 8. An illustrating example for various resolutions of $\bar{d}$ (points in different clusters are marked with different colors).



(a) The (geographic) distance measure

(b) The latency measure

Fig. 9. Clustering for the WonderNetwork dataset: (a) the (geographic) distance measure and (b) the latency measure.

they are marked with different colors in Figure 9(b) as there are no directly connected cables between New Delhi and Lahore and that leads to large latency between these two servers.

6 Conclusion

In this paper, we proposed using the exponentially twisted sampling along with path measures for centrality analysis and community detection in attributed networks. For signed networks, we defined the influence centralities by using a path measure from opinions dynamics and the trust centralities by using a path measure from a chain of trust. For attributed networks with node attributes, we also defined advertisement-specific influence centralities by using a specific path measure that models influence cascades in such networks. For a network with a distance measure, we defined the path measure as the total distance along a path. By specifying the desired average distance between two randomly sampled nodes, we showed how one can detect communities with various resolution parameters. Various experiments were conducted to illustrate these exponentially twisted sampling methods.

The goal of this paper is to provide a general framework for centrality analysis and community detection in attributed networks. We do not aim to solve a specific ranking task or a community detection task for a specific dataset. To apply our framework for a specific ranking task or a community detection task for a specific dataset, one needs to carefully choose the path measures and the desired average values of the path measures. This might require lots of fine-tuning of parameters and it is beyond the scope of this paper. There are various practical applications of network sampling and community detection, e.g., the topic recognition in academic networks [50], and the network mapping problem [51]. Finding the suitable viewpoints definitely requires further insight for each of these applications. Thus, the more we understand about the application, the better the path measures and the desired average values of the path measures can be assigned for that application.

References

[1] L. C. Freeman, "A set of measures of centrality based on betweenness," *Sociometry*, pp. 35–41, 1977.

[2] ——, "Centrality in social networks conceptual clarification," *Social networks*, vol. 1, no. 3, pp. 215–239, 1979.

[3] M. Girvan and M. E. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences*, vol. 99, no. 12, pp. 7821–7826, 2002.

[4] A. Clauset, M. E. Newman, and C. Moore, "Finding community structure in very large networks," *Physical review E*, vol. 70, no. 6, p. 066111, 2004.

[5] M. Newman, *Networks: an introduction*. OUP Oxford, 2009.

[6] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Computer networks and ISDN systems*, vol. 30, no. 1, pp. 107–117, 1998.

[7] T. H. Haveliwala, "Topic-sensitive pagerank," in *Proceedings of the 11th international conference on World Wide Web*. ACM, 2002, pp. 517–526.

[8] G. Jeh and J. Widom, "Scaling personalized web search," in *Proceedings of the 12th international conference on World Wide Web*. ACM, 2003, pp. 271–279.

[9] C.-S. Chang, C.-J. Chang, W.-T. Hsieh, D.-S. Lee, L.-H. Liou, and W. Liao, "Relative centrality and local community detection," *Network Science*, vol. 3, no. 4, pp. 445–479, 2015.

[10] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi, "Defining and identifying communities in networks," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, no. 9, pp. 2658–2663, 2004.

[11] Y. Hu, H. Chen, P. Zhang, M. Li, Z. Di, and Y. Fan, "Comparative definition of community and corresponding identifying algorithm," *Physical Review E*, vol. 78, no. 2, p. 026121, 2008.

[12] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3, pp. 75–174, 2010.

[13] J. Leskovec, K. J. Lang, and M. Mahoney, "Empirical comparison of algorithms for network community detection," in *Proceedings of the 19th international conference on World wide web*. ACM, 2010, pp. 631–640.

[14] 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*. ACM, 2012, p. 3.

[15] M. E. Newman, "Fast algorithm for detecting community structure in networks," *Physical review E*, vol. 69, no. 6, p. 066133, 2004.

[16] P. J. Mucha, T. Richardson, K. Macon, M. A. Porter, and J.-P. Onnela, "Community structure in time-dependent, multiscale, and multiplex networks," *Science*, vol. 328, no. 5980, pp. 876–878, 2010.

[17] M. Rosvall and C. T. Bergstrom, "An information-theoretic framework for resolving community structure in complex networks," *Proceedings of the National Academy of Sciences*, vol. 104, no. 18, pp. 7327–7331, 2007.

[18] ——, "Maps of random walks on complex networks reveal community structure," *Proceedings of the National Academy of Sciences*, vol. 105, no. 4, pp. 1118–1123, 2008.

[19] R. Lambiotte, "Multi-scale modularity in complex networks," in *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt), 2010 Proceedings of the 8th International Symposium on*. IEEE, 2010, pp. 546–553.

[20] J.-C. Delvenne, S. N. Yaliraki, and M. Barahona, "Stability of graph communities across time scales," *Proceedings of the National Academy of Sciences*, vol. 107, no. 29, pp. 12 755–12 760, 2010.

[21] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Gossip algorithms: Design, analysis and applications," in *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, vol. 3. IEEE, 2005, pp. 1653–1664.

[22] A. K. Jain, "Data clustering: 50 years beyond k-means," *Pattern Recognition Letters*, vol. 31, no. 8, pp. 651–666, 2010.

[23] C.-S. Chang, D.-S. Lee, L.-H. Liou, S.-M. Lu, and M.-H. Wu, "A probabilistic framework for structural analysis and community detection in directed networks," *IEEE/ACM Transactions on Networking*, 2017.

[24] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a feather: Homophily in social networks," *Annual review of sociology*, vol. 27, no. 1, pp. 415–444, 2001.

[25] J. Leskovec, D. Huttenlocher, and J. Kleinberg, "Predicting positive and negative links in online social networks," in *Proceedings of the 19th international conference on World wide web*. ACM, 2010, pp. 641–650.

[26] J. Weng, E.-P. Lim, J. Jiang, and Q. He, "Twitterrank: finding topic-sensitive influential twitterers," in *Proceedings of the third ACM international conference on Web search and data mining*. ACM, 2010, pp. 261–270.

[27] J. Yang, J. McAuley, and J. Leskovec, "Community detection in networks with node attributes," in *Data Mining (ICDM), 2013 IEEE 13th international conference on*. IEEE, 2013, pp. 1151–1156.

[28] N. Z. Gong, A. Talwalkar, L. Mackey, L. Huang, E. C. R. Shin, E. Stefanov, E. R. Shi, and D. Song, "Joint link prediction and attribute inference using a social-attribute network," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 5, no. 2, p. 27, 2014.

[29] S. Wang, C. Aggarwal, J. Tang, and H. Liu, "Attributed signed network embedding," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. ACM, 2017, pp. 137–146.

[30] X. Huang, J. Li, and X. Hu, "Label informed attributed network embedding," in *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*. ACM, 2017, pp. 731–739.

[31] C.-S. Chang, C.-Y. Hsu, J. Cheng, and D.-S. Lee, "A general probabilistic framework for detecting community structure in networks," in *IEEE INFOCOM '11*, April 2011.

[32] S. Juneja and P. Shahabuddin, "Rare-event simulation techniques: an introduction and recent advances," *Handbooks in operations research and management science*, vol. 13, pp. 291–350, 2006.

[33] T. M. Cover and J. A. Thomas, *Elements of information theory*. John Wiley & Sons, 2012.

[34] C.-S. Chang, *Performance guarantees in communication networks*. Springer Science & Business Media, 2012.

[35] L. A. Adamic and N. Glance, "The political blogosphere and the 2004 us election: divided they blog," in *Proceedings of the 3rd international workshop on Link discovery*. ACM, 2005, pp. 36–43.

[36] J. Leskovec, L. Backstrom, and J. Kleinberg, "Memetracker data," 2008.

[37] "The wondernetwork dataset," https://wondernetwork.com/.

[38] R. Nelson, *Probability, stochastic processes, and queueing theory: the mathematics of computer performance modeling*. Springer Verlag, 1995.

[39] U. Brandes, D. Delling, M. Gaertler, R. Görke, M. Hoefer, Z. Nikoloski, and D. Wagner, "Maximizing modularity is hard," *arXiv preprint physics/0608255*, 2006.

[40] M. E. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Physical review E*, vol. 74, no. 3, p. 036104, 2006.

[41] C.-S. Chang, W. Liao, Y.-S. Chen, and L.-H. Liou, "A mathematical theory for clustering in metric spaces," *IEEE Transactions on Network Science and Engineering*, vol. 3, no. 1, pp. 2–16, 2016.

[42] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, p. P10008, 2008.

[43] A. Y. Ng, M. I. Jordan, Y. Weiss *et al.*, "On spectral clustering: Analysis and an algorithm," *Advances in neural information processing systems*, vol. 2, pp. 849–856, 2002.

[44] A. Ben-Hur, D. Horn, H. T. Siegelmann, and V. Vapnik, "Support vector clustering," *The Journal of Machine Learning Research*, vol. 2, pp. 125–137, 2002.

[45] C.-S. Chang, C.-T. Chang, D.-S. Lee, and L.-H. Liou, "K-sets+: a linear-time clustering algorithm for data points with a sparse similarity measure," in *The 3rd IEEE Int'l Conf. on Cloud and Big Data Computing (CBDCom)*, August 2017.

[46] C.-H. Chang and C.-S. Chang, "Exponentially twisted sampling: a unified approach for centrality analysis in attributed networks," *arXiv preprint arXiv:1708.00379*, 2017.

[47] S. Osiński and D. Weiss, "Carrot2: Design of a flexible and efficient web information retrieval framework," in *International atlantic web intelligence conference*. Springer, 2005, pp. 439–444.

[48] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM computing surveys (CSUR)*, vol. 31, no. 3, pp. 264–323, 1999.

[49] "The submarine cable map," http://www.submarinecablemap.com/.

[50] X. Huang, C.-a. Chen, C. Peng, X. Wu, L. Fu, and X. Wang, "Topic-sensitive influential paper discovery in citation network," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 2018, pp. 16–28.

[51] L. Fu, X. Wu, Z. Hu, X. Fu, and X. Wang, "De-anonymizing social networks with overlapping community structure," *arXiv preprint arXiv:1712.04282*, 2017.

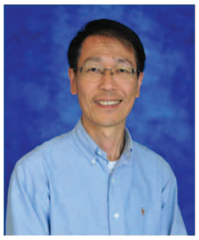
Cheng-Hsun Chang received his M.S. degree in the Institute of Communications Engineering, National Tsing-Hua University, Hsinchu, Taiwan, in 2017. He is currently a firmware engineer at Phison Electronics Corporatio, Miaoli, Taiwan, R.O.C., developing SSD firmware code.

Cheng-Shang Chang (S'85-M'86-M'89-SM'93-F'04) received the B.S. degree from National Taiwan University, Taipei, Taiwan, in 1983, and the M.S. and Ph.D. degrees from Columbia University, New York, NY, USA, in 1986 and 1989, respectively, all in electrical engineering.

From 1989 to 1993, he was employed as a Research Staff Member with the IBM Thomas J. Watson Research Center, Yorktown Heights, NY, USA. Since 1993, he has been with the Department of Electrical Engineering, National Tsing Hua University, Taiwan, where he is a Tsing Hua Distinguished Chair Professor. He is the author of the book Performance Guarantees in Communication Networks (Springer, 2000) and the coauthor of the book Principles, Architectures and Mathematical Theory of High Performance Packet Switches (Ministry of Education, R.O.C., 2006). His current research interests are concerned with network science, big data analytics, mathematical modeling of the Internet, and high-speed switching.

Dr. Chang served as an Editor for Operations Research from 1992 to 1999, an Editor for the *IEEE/ACM TRANSACTIONS ON NETWORKING* from 2007 to 2009, and an Editor for the *IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING* from 2014 to 2017. He is currently serving as an Editor-at-Large for the *IEEE/ACM TRANSACTIONS ON NETWORKING*. He is a member of IFIP Working Group 7.3. He received an IBM Outstanding Innovation Award in 1992, an IBM Faculty Partnership Award in 2001, and Outstanding Research Awards from the National Science Council, Taiwan, in 1998, 2000, and 2002, respectively. He also received Outstanding Teaching Awards from both the College of EECS and the university itself in 2003. He was appointed as the first Y. Z. Hsu Scientific Chair Professor in 2002. He received the Merit NSC Research Fellow Award from the National Science Council, R.O.C. in 2011. He also received the Academic Award in 2011 and the National Chair Professorship in 2017 from the Ministry of Education, R.O.C. He is the recipient of the 2017 IEEE INFOCOM Achievement Award.

Chia-Tai Chang received his M.S. degree in the Institute of Communications Engineering, National Tsing-Hua University, Hsinchu, Taiwan, in 2017. He is currently a system developer engineer at Realtek Semiconductor Corporation, Hsinchu, Taiwan, R.O.C., developing color processing algorithms and maintaining display color function in monitor.



Duan-Shin Lee (S'89-M'90-SM'98) received the B.S. degree from National Tsing Hua University, Taiwan, in 1983, and the MS and Ph.D. degrees from Columbia University, New York, in 1987 and 1990, all in electrical engineering. He worked as a research staff member at the C&C Research Laboratory of NEC USA, Inc. in Princeton, New Jersey from 1990 to 1998. He joined the Department of Computer Science of National Tsing Hua University in Hsinchu, Taiwan, in 1998. Since August 2003, he has been a professor. He received a best paper award from the Y.Z. Hsu Foundation in 2006. He served as an editor for the Journal of Information Science and Engineering between 2013 and 2015. He is currently an editor for Performance Evaluation. Dr. Lee's current research interests are network science, game theory, machine learning and high-speed networks. He is a senior IEEE member.

Ping-En Lu received his B.S. degree in communication engineering from the Yuan Ze University, Taoyuan, Taiwan, in 2015. He is currently pursuing the Ph.D. degree in the Institute of Communications Engineering, National Tsing-Hua University. His research interest is in efficient clustering algorithms and deep learning algorithms.