Knowl Inf Syst (2019) 58:701–728
https://doi.org/10.1007/s10115-018-1167-0

REGULAR PAPER

Practical characterization of large networks using neighborhood information

Pinghui Wang[1,2] · Junzhou Zhao[3] · Bruno Ribeiro[4] · John C. S. Lui[5] · Don Towsley[6] · Xiaohong Guan[1,7]

Received: 12 January 2017 / Revised: 7 January 2018 / Accepted: 30 January 2018 /
Published online: 14 February 2018
© Springer-Verlag London Ltd., part of Springer Nature 2018

Abstract Characterizing large complex networks such as online social networks through node querying is a challenging task. Network service providers often impose severe constraints on the query rate, hence limiting the sample size to a small fraction of the total network of interest. Various ad hoc subgraph sampling methods have been proposed, but many of them give biased estimates and no theoretical basis on the accuracy. In this work, we focus on developing sampling methods for large networks where querying a node also reveals partial structural information about its neighbors. Our methods are optimized for

✉ Junzhou Zhao
junzhouzhao@gmail.com

Pinghui Wang
phwang@mail.xjtu.edu.cn

Bruno Ribeiro
ribeiro@cs.purdue.edu

John C. S. Lui
cslui@cse.cuhk.edu

Don Towsley
towsley@cs.umass.edu

Xiaohong Guan
xhguan@mail.xjtu.edu.cn

1 MOE Key Laboratory for Intelligent Networks and Network Security, Xi'an Jiaotong University, Xi'an, China

2 Shenzhen Research Institute of Xi'an Jiaotong University, Shenzhen, China

3 Division of Computer, Electrical and Mathematical Sciences and Engineering, King Abdullah University of Science and Technology, Thuwal, Saudi Arabia

4 School of Computer Science, Purdue University, West Lafayette, IN, USA

5 Department of Computer Science and Engineering, The Chinese University of Hong Kong, Sha Tin, Hong Kong

6 Department of Computer Science, University of Massachusetts Amherst, Amherst, MA, USA

7 Center for Intelligent and Networked Systems, Tsinghua University, Beijing, China

NoSQL graph databases (if the database can be accessed directly), or utilize Web APIs available on most major large networks for graph sampling. We show that our sampling method has provable convergence guarantees on being an unbiased estimator, and it is more accurate than state-of-the-art methods. We also explore methods to uncover shortest paths between a subset of nodes and detect high degree nodes by sampling only a small fraction of the network of interest. Our results demonstrate that utilizing neighborhood information yields methods that are two orders of magnitude faster than state-of-the-art methods.

Keywords Crawling · Graph sampling · Online social network · Random walk

1 Introduction

The literature on sampling large networks is vast and rich. Various techniques have been proposed for subgraph sampling and characterization of large networks [1–3] (refer to Ahmed et al. [4] for a good survey). These techniques, however, often lack provable guarantees. This means that after sampling a fraction of a large network, one has no guarantees whether the metrics obtained are to be trusted. Fortunately, researchers have recently made a push toward network characterization through sampling with provable properties and accuracy guarantees.

Techniques adapted to sample networks stored at NoSQL graph databases or accessible from Web APIs (e.g. available on Facebook,[1] Sina microblog,[2] Quora,[3] and CiteSeerX[4]) must refrain from randomly sampling too many nodes and all together avoid sampling edges, either due to caching inefficiencies or limitations in the API. In practice, most large networks such as online social networks (OSNs), including those we present in this study, do not provide random sampling primitives. Practitioners perform random sampling by guessing user IDs in the user ID space, which, if sparsely populated, imposes a large number of query misses until a valid user is found. In this context, techniques that heavily rely on random sampling, such as Dasgupta et al. [5], suffer from the low query rate. Dasgupta et al. [5] partially compensate the low query rate through the use of neighborhood information present in the node query reply. Similarly, graph streaming techniques, such as Ahmed et al. [4], are also not well adapted to this environment as they require visiting all edges, which is prohibitively expensive for large networks with millions or even billions of edges.

Recently, great focus has been placed on developing techniques that use specially constructed "*crawlers*" to query large networks and to provide asymptotically unbiased estimates of a handful of network characteristics [6,7]. Chief among these techniques are random walks (RWs), which provide provable accuracy and convergence guarantees (see Ribeiro and Towsley [8] and Avrechenkov et al. [9]). RWs present a number of desirable properties that are useful to characterize large networks: (1) they require either few or no independently sampled nodes and produce asymptotically unbiased estimates and accuracy guarantees under mild conditions for a large family of directed[5] and undirected networks, even when the network of interest has multiple disconnected components, as long as some limited amounts of random sampling are available [6,8,9]; (2) they use crawling to collect samples (which

[1] http://www.facebook.com.

[2] http://www.weibo.com.

[3] http://www.quora.com.

[4] http://citeseerx.ist.psu.edu.

[5] In directed networks where querying a node retrieves the node¡¯s incoming and outgoing edges.

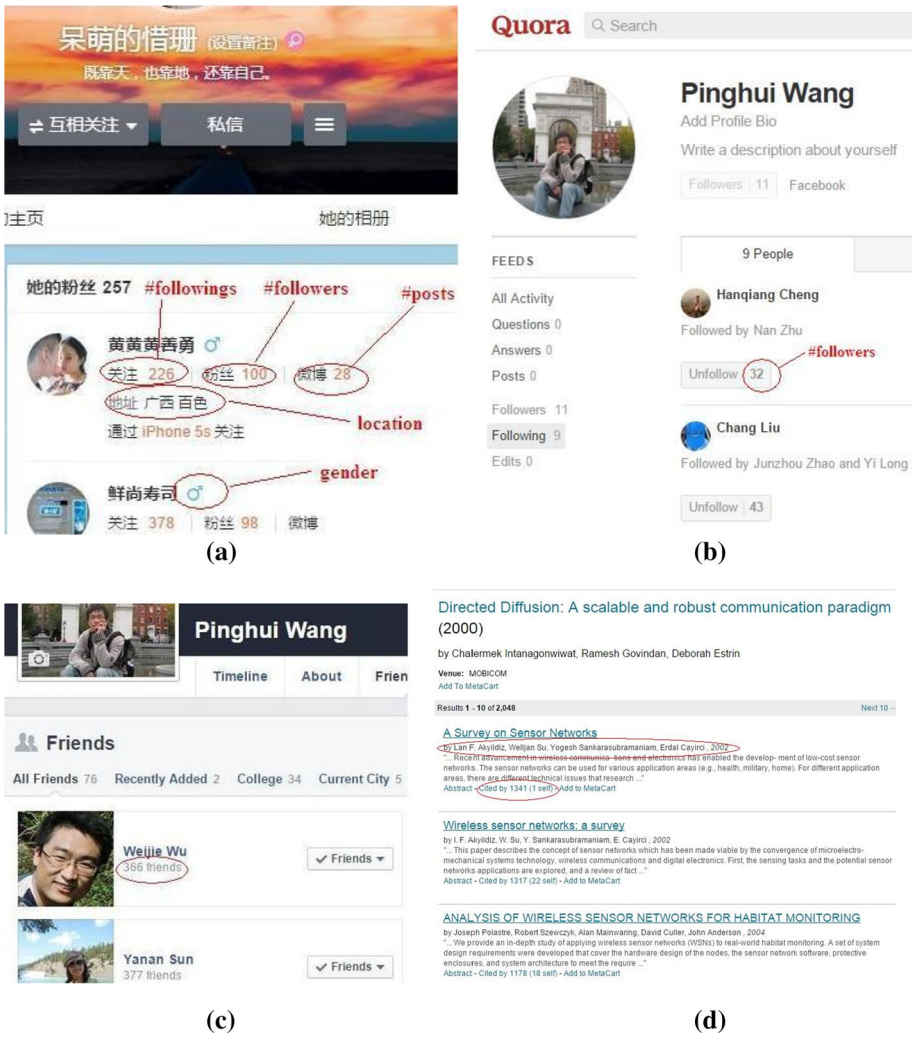


Fig. 1 Examples of networks providing neighborhood information with no sampling cost. **a** Sina microblog (directed graph), **b** Quora (directed graph), **c** Facebook (undirected graph), **d** Citeseerx (directed graph)

effectively implements importance sampling on node degrees) and can achieve relatively high query rates on NoSQL graph databases or using Web APIs; (3) they do not require any advance knowledge of the network, such as its size or topology. However, existing RW techniques do not take advantage of the extra neighborhood information, despite the fact that neighborhood information is readily available in many networks at (practically) no sampling cost (obtained from the node query reply). Several examples are given in Fig. 1. When we crawl a user's profile on Sina microblog, we obtain its neighbors' information such as the number of followers, the number of followings, location, and gender. When we crawl a user's profile on Facebook, we obtain its friend list and its friends' friend counts. When we crawl a user's profile on Quora (a question-and-answer website), we obtain its neighbors (followers and followings) and the number of followers of each of its neighbors. When we search a

paper on CiteSeerX, we obtain the information of its citations and references such as the authors and the number of citations. To efficiently support the above functions, we guess that these network service providers organize and store each node's neighbors and its neighbors' brief summaries separately, which can be easily achieved by NoSQL databases. The extra neighborhood information is obtained with little sampling cost in the above examples. When the node and edge labels of interest are included in the extra neighborhood information, clearly, we can reduce the sampling cost by utilizing this extra information to characterize node and edge labels. However, including such extra information in RW-based estimator while retaining unbiased guarantees is challenging due to different types of biases involved in the sampling process. Moreover, in this paper, we observe that other applications such as high degree node detection and shortest path discovery can also benefit from the extra neighborhood information.

Contributions In this work, we consider the generalization of RW sampling and combine current state-of-the-art estimators to include neighborhood information for estimating node and edge label densities. Our estimator drastically reduces (by 4-fold) the number of samples required to achieve the same estimation accuracy. Examples of OSNs that provide neighborhood information are found everywhere, e.g. Facebook, Sina microblog, and Google Scholar.[6] Our generalization allows us to include neighborhood information in the estimation of a variety of network characteristics from nodes sampled using a RW-based technique called Frontier Sampling [6]. We also explore methods to uncover shortest paths between a subset of nodes and detect high degree nodes by sampling only a small fraction of the network of interest. Our results show that utilizing neighborhood information yields methods that are two orders of magnitude faster than state-of-the-art methods.

This paper is organized as follows. Several basic crawling techniques are summarized in Sect. 2. In Sect. 3, we present the methodology of using neighborhood information to estimate node label density. In Sects. 4, 5, and 6, we propose methods using neighborhood information to estimate edge label density, detect high degree nodes, and uncover shortest paths respectively. The performance evaluation and testing results are presented in Sect. 7. Section 8 summarizes the related work. Concluding remarks then follow.

2 Preliminaries

In this section we present three basic graph sampling methods: *Uniform Vertex Sampling* (UNI), *Random Walk* (RW) [11], and *Frontier Sampling* (FS) [6], which are underlying techniques for problems discussed in later sections. For ease of presentation, in this section, we only present methods for undirected graphs. One way to convert a directed graph into an undirected graph is by ignoring the direction of edges. Unless we state otherwise, denote by $G_{\mathrm{d}} = (V, E_{\mathrm{d}})$ the directed graph under study, and $G = (V, E)$ the undirected graph generated by ignoring the direction of edges in G_{d}. For ease of reading, we list notation used throughout the paper in Table 1.

2.1 Uniform node sampling (UNI)

UNI is the simplest method to provide unbiased estimates of population estimates, where each node $v \in V$ is sampled with the same probability

[6] http://scholar.google.com.

Table 1 Table of notation

$G = (V, E)$	Undirected graph
$G_{\mathrm{d}} = (V, E_{\mathrm{d}})$	Directed graph
$\mathcal{N}_v$	The set of neighbors of node v in G
$\mathcal{N}_v^{(\mathrm{I})}$	The set of followers of node v in G_{d}
$\mathcal{N}_v^{(\mathrm{O})}$	The set of followings of node v in G_{d}
$d_v, d_v^{(\mathrm{I})}, d_v^{(\mathrm{o})}$	Degree, in-degree, out-degree of node v respectively
$\{l_1, \ldots, l_K\}$	The set of node labels
$\{l'_1, \ldots, l'_{K'}\}$	The set of edge labels
L_v	The label of node v
$L_{u,v}$	The label of edge (u, v)
$\boldsymbol{\theta} = (\theta_1, \ldots, \theta_K)$	Node label density
$\boldsymbol{\tau} = (\tau_1, \ldots, \tau_{K'})$	Edge label density
$[s_i]_{i=1,\ldots,n}$	List of nodes sampled by UNI, RW, and FS
$[(s_i^-, s_i)]_{i=1,\ldots,n}$	List of edges sampled by RW and FS
	$s_i^- = s_{i-1}$ holds for RW but does not always hold for FS
n	The number of sampled nodes
K, K'	The number of node labels and edge labels, respectively
$\pi_v^{\mathrm{UNI}}, \pi_v^{\mathrm{RW}}, \pi_v^{\mathrm{FS}}$	The stationary probability of sampling node v at each step of UNI, RW, and FS respectively

$$\pi_v^{\mathrm{UNI}} = \frac{1}{|V|}.$$

There are few OSNs that provide APIs support for UNI. One website that supports UNI is Wikipedia, where one can query a randomly sampled Wikipedia page. On networks such as Facebook, Foursquare, Flickr,[7] Sina microblog, and Xiami,[8] one can sample users (nodes) as users have numeric IDs between the minimum and the maximum ID values. Unfortunately, ID values of users in many networks (e.g. Flickr, Facebook, and Sina microblog) are not assigned sequentially, and the ID space is sparsely populated [6,12]. Hence, a randomly generated ID may not correspond to a valid user and a considerable computational effort may be required to generate a legitimate ID. To sample a large number of nodes, therefore, UNI is only practical on networks that provide the API and those whose user ID space is densely packed.

2.2 Random walk (RW)

RW has been extensively studied in the graph theory literature [11]. From an initial node, a walker selects a neighbor at random as the next-hop node. The walker moves to this neighbor and repeats the process. Denote by $\mathcal{N}_u$ the set of neighbors of any node $u \in V$. Let d_u (i.e., $d_u = |\mathcal{N}_u|$) be the degree of u. Formally, a RW can be viewed as a Markov chain

[7] http://www.flickr.com.

[8] http://www.xiami.com.

with transition matrix $[P^{\mathrm{RW}}_{u,v}]_{u,v\in V}$, where $P^{\mathrm{RW}}_{u,v}$ is defined as the probability of node v being selected as the next-hop node given that the walker is currently at node u. Formally, we have

$$P^{\mathrm{RW}}_{u,v} = \begin{cases} 1/d_u & \text{if } v \in \mathcal{N}_u, \\ 0 & \text{otherwise.} \end{cases}$$

The stationary distribution $\boldsymbol{\pi}^{\mathrm{RW}}$ of this Markov chain is

$$\pi^{\mathrm{RW}}_v = \frac{d_v}{2|E|}, \quad v \in V.$$

For a connected and non-bipartite graph G, the probability of being at node v converges to the above stationary distribution [11]. In addition, RW can also be used to sample edges randomly, and the probability of traversing each edge converges to the same value $\frac{1}{|E|}$ [6].

2.3 Frontier sampling (FS)

FS [6] is a fully distributed sampling algorithm that performs m independent RWs on G. If $m = 1$, FS behaves exactly like a RW. When $m > 1$, compared to a single RW, FS is more robust to the problem that arises from the walker getting trapped at a loosely connected component of G. Each FS walker has a predefined budget B (we explain how B is chosen at the end of this section). At each step, an FS walker at node u moves to a node randomly selected from $\mathcal{N}_u$, deducting from the budget B a random quantity $X \sim \mathrm{Exp}(d_u)$, an exponentially distributed random variable with mean $1/d_u$. FS stops when B becomes negative. If G is a connected and non-bipartite graph, the probability that a node v is sampled by FS converges to the following distribution

$$\pi^{\mathrm{FS}}_v = \frac{d_v}{2|E|}, \quad v \in V.$$

FS can also be used to sample edges randomly, as the probability of traversing each edge converges to the same value $\frac{1}{|E|}$ [6]. Let n denote the number of nodes that one wishes to sample. Define $\bar{d} = \frac{\sum_{v\in V} d_v}{|V|}$ as the average degree. The choice of budget B is often defined as $\bar{d}n/m$. In practice, one does not need to know $\bar{d}$ as B may be increased dynamically on-the-fly. Because we can adjust B on-the-fly, in what follows we take the liberty to assume that FS samples exactly n nodes.

In the following sections, we assume that graph G is connected and non-bipartite. Let $[s_i]_{1\le i\le n}$ be the list (or, sequence) of sampled nodes by UNI, RW, and FS, where s_i is the ith sampled node. Note that $s_1, \ldots, s_n$ are not necessary different from each other. Ref. [6] reveals that RW and FS are also effective for randomly sampling edges. Let $[(s_i^-, s_i)]_{1\le i\le n}$ be the list of sampled edges by RW and FS, where (s_i^-, s_i) is the ith sampled edge, i.e., a RW (or FS) walker moved from node s_i^- to s_i. For RW, we easily have $s_i^- = s_{i-1}$, where s_0 is the initial node of RW. As we mentioned, FS has m independent walkers and s_{i-1} and s_i may not be generated by the same walker, so s_i^- does not necessarily equal s_{i-1}. Let "a.s." denote "almost sure" converge, i.e., the event of interest happens with probability one. Then we have

Lemma 1 [6,13,14] *For any function $\phi(v) : V \to \mathbb{R}$, where $\sum_{v\in V} \phi(v) < \infty$, we have the following equation for UNI, RW, and FS.*

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \phi(s_i) \xrightarrow{a.s.} \sum_{v\in V} \phi(v)\pi_v.$$

3 Node label density estimation

In this section, we present methods for estimating node label density. Formally, denote by L_v the label of node $v \in V$, with range $\{l_1, \ldots, l_K\}$. Define

$$\theta_k = \frac{|\{v : L_v = l_k \wedge v \in V\}|}{|V|}, \quad 1 \le k \le K,$$

i.e., the fraction of nodes with label l_k. Then, the node label density is defined as $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_K)$. For example, when L_v is defined as the degree of node v, then $\boldsymbol{\theta}$ is the degree distribution of nodes in V. When L_v is defined as the gender of node (or user) v, then $\boldsymbol{\theta}$ is the gender distribution of nodes in V.

3.1 Simple estimators of node densities

To estimate $\boldsymbol{\theta}$ based on sampled nodes $[s_i]_{1 \le i \le n}$, the stationary distribution of sampling methods (e.g. UNI, RW, and FS) $\boldsymbol{\pi} = [\pi_v]_{v \in V}$ is needed to correct the bias induced by the underlying sampling method. For $v \in V$, we have $\pi_v = \frac{1}{|V|}$ for UNI, and $\pi_v = \frac{d_v}{2|E|}$ for RW and FS. Since the values of $|V|$ and $|E|$ are usually unknown, we cannot correct the sampling bias in a direct manner. Instead, one may use a non-normalized stationary distribution $\hat{\boldsymbol{\pi}} = [\hat{\pi}_v]_{v \in V}$ to reweigh sampled nodes, where $\hat{\pi}_v$ is computed as

$$\hat{\pi}_v = \begin{cases} 1 & \text{for UNI,} \\ d_v & \text{for RW and FS.} \end{cases} \tag{1}$$

We easily find that $\pi_v \propto \hat{\pi}_v$. Let $\mathbf{1}(\mathbf{P})$ denote the indicator function that equals one when predicate $\mathbf{P}$ is true, and zero otherwise. Similar to the Horvitz–Thompson estimator [16], we use inverse probability weighting to reweigh sampled nodes. That is, we estimate θ_k as

$$\hat{\theta}_k = \frac{1}{C} \sum_{i=1}^{n} \frac{\mathbf{1}(L_{s_i} = l_k)}{\hat{\pi}_{s_i}}, \quad 1 \le k \le K, \tag{2}$$

where $C = \sum_{i=1}^{n} \hat{\pi}_{s_i}^{-1}$. Ribeiro and Towsley [6] show that $\hat{\theta}_k$ is an asymptotically unbiased estimate of θ_k.

3.2 Estimators using neighborhood information of sampled nodes

A node $w \in V$ appears in d_w nodes' neighbor sets. When the degrees and the node labels of sampled nodes' neighbors are available, therefore, nodes with larger degrees have a larger chance to appear as sampled nodes' neighbors in comparison with nodes with smaller degrees even for UNI. Similar to the Horvitz–Thompson estimator, we propose the following estimator utilizing the free neighborhood information

$$\breve{\theta}_k = \frac{1}{\breve{C}} \sum_{i=1}^{n} \sum_{w \in \mathcal{N}_{s_i}} \frac{\mathbf{1}(L_w = l_k)}{\hat{\pi}_{s_i} d_w}, \quad 1 \le k \le K, \tag{3}$$

where $\breve{C} = \sum_{i=1}^{n} \sum_{w \in \mathcal{N}_{s_i}} \hat{\pi}_{s_i}^{-1} d_w^{-1}$. The above estimator is similar to one proposed in Dasgupta et al. [5]. However, the estimator in Dasgupta et al. [5] requires $|V|$ to be known in advance, which is usually not available. Moreover, Dasgupta et al. [5] focus on designing independent node sampling methods (e.g. UNI, independent weighted node sampling), which we argued has a low query rate. Whereas we focus on crawling methods such as RW and FS.

For any $v \in V$, Eq. (1) shows that $\frac{\pi_v}{\hat{\pi}_v}$ has the same value, denoted as C_π. We have $C_\pi = \frac{1}{2|E|}$ for RW and FS and $C_\pi = \frac{1}{|V|}$ for UNI. Next, we analyze the accuracy of estimator $\breve{\theta}_k$.

Theorem 1 *$\breve{\theta}_k$ is an asymptotically unbiased estimate of θ_k.*

Proof Let $\phi(v) = \sum_{w \in \mathcal{N}_v} \frac{\mathbf{1}(L_w = l_k)}{\hat{\pi}_v d_w}$. Applying Lemma 1, we have

$$\begin{aligned}
&\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \sum_{w \in \mathcal{N}_{s_i}} \frac{\mathbf{1}(L_w = l_k)}{\hat{\pi}_{s_i} d_w} = \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \phi(s_i) \\
&\xrightarrow{a.s.} \sum_{v \in V} \pi_v \phi(v) \\
&= C_\pi \sum_{w \in V} \sum_{v \in \mathcal{N}_w} \frac{\mathbf{1}(L_w = l_k)}{d_w} \\
&= C_\pi \sum_{w \in V} \mathbf{1}(L_w = l_k) \\
&= C_\pi |V| \theta_k.
\end{aligned}$$

Similarly, we have $\lim_{n\to\infty} \frac{\breve{C}}{n} \xrightarrow{a.s.} C_\pi |V|$. Thus, we obtain $\lim_{n\to\infty} \breve{\theta}_k \xrightarrow{a.s.} \theta_k$. □

$\breve{\theta}_k$ is computed based on node labels of sampled nodes' neighbors, while $\hat{\theta}_k$ is computed based on node labels of sampled nodes. We can easily find that neighbors of sampled nodes are biased to nodes with high degrees even for UNI. Therefore, $\breve{\theta}_k$ is an estimator based on biased samples. Ribeiro and Towsley [6] show that UNI has a smaller error for characterizing small degree nodes than biased sampling methods such as FS. It is consistent with our results in Sect. 7, which show that $\breve{\theta}_k$ may exhibit a larger error than $\hat{\theta}_k$ defined in (2). Ref. [10] gives an optimal method to combine unbiased estimators. According to this method, we present the following mixture estimator for θ_k

$$\hat{\theta}_k^{\text{mix}} = \alpha_k \hat{\theta}_k + (1 - \alpha_k)\breve{\theta}_k, \quad 1 \le k \le K, \tag{4}$$

where parameter α_k is defined as

$$\alpha_k = \frac{\text{Var}(\breve{\theta}_k)}{\text{Var}(\hat{\theta}_k) + \text{Var}(\breve{\theta}_k)}.$$

α_k lies between zero and one, and is used to determine the relative importance of two estimates $\hat{\theta}_k$ and $\breve{\theta}_k$. When $\hat{\theta}_k$ and $\breve{\theta}_k$ are independent, $\hat{\theta}_k^{\text{mix}}$ has the smallest variance $\frac{\text{Var}(\hat{\theta}_k)\text{Var}(\breve{\theta}_k)}{\text{Var}(\hat{\theta}_k)+\text{Var}(\breve{\theta}_k)}$. In practice, $\hat{\theta}_k$ and $\breve{\theta}_k$ might not be independent, our later experimental results show that $\hat{\theta}_k^{\text{mix}}$ exhibits smaller errors than $\hat{\theta}_k$ and $\breve{\theta}_k$ for many real networks.

In what follows we propose an estimator of $\boldsymbol{\theta}$ using the available neighborhood information of sampled nodes for directed OSNs such as Sina microblog, where a node has knowledge of in-degrees (i.e., the number of followers) and out-degrees (i.e., the number of followings) of its incoming neighbors and outgoing neighbors. For a node $v \in V$, denote by $d_v^{(\text{I})}$ its in-degree and $d_v^{(\text{O})}$ its out-degree. Let

$$\mathcal{N}_v^{(\text{I})} = \{u : (u, v) \in E_{\text{d}}\}$$

be the set of followers of v, and

$$\mathcal{N}_v^{(\text{O})} = \{u : (v, u) \in E_{\text{d}}\}$$

be the set of followings of v. Let $\mathcal{N}_v = \mathcal{N}_v^{(\mathrm{I})} \cup \mathcal{N}_v^{(\mathrm{O})}$. Define

$$\psi_{u,v} = \begin{cases} 2, & (u,v) \in E_{\mathrm{d}} \wedge (v,u) \in E_{\mathrm{d}} \\ 0, & (u,v) \notin E_{\mathrm{d}} \wedge (v,u) \notin E_{\mathrm{d}} \\ 1, & \text{otherwise.} \end{cases}$$

A node $w \in V$ appears as a following of $d_w^{(\mathrm{I})}$ nodes in $\mathcal{N}_w^{(\mathrm{I})}$ and as a follower of $d_w^{(\mathrm{O})}$ nodes in $\mathcal{N}_w^{(\mathrm{O})}$. Therefore, node w with larger $d_w^{(\mathrm{I})} + d_w^{(\mathrm{O})}$ has a larger chance to appear as a neighbor of sampled nodes even for UNI. Similar to $\breve{\theta}_k$, using sampled nodes' followers and followings we estimate θ_k as

$$\breve{\theta}_k^* = \frac{1}{\breve{C}_{\mathrm{d}}} \sum_{i=1}^{n} \sum_{w \in \mathcal{N}_{s_i}} \frac{\psi_{s_i,w} \mathbf{1}(L_w = l_k)}{\hat{\pi}_{s_i}(d_w^{(\mathrm{I})} + d_w^{(\mathrm{O})})}, \quad 1 \le k \le K,$$

where $\breve{C}_{\mathrm{d}} = \sum_{i=1}^{n} \sum_{w \in \mathcal{N}_{s_i}} \psi_{s_i,w} \hat{\pi}_{s_i}^{-1} (d_w^{(\mathrm{I})} + d_w^{(\mathrm{O})})^{-1}$.

Theorem 2 *$\breve{\theta}_k^*$ is an asymptotically unbiased estimate of θ_k.*

Proof Let $\phi_v = \sum_{w \in \mathcal{N}_v} \frac{\psi_{v,w} \mathbf{1}(L_w = l_k)}{\hat{\pi}_v (d_w^{(\mathrm{I})} + d_w^{(\mathrm{O})})}$. Applying Lemma 1, we have

$$\begin{aligned}
&\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \sum_{w \in \mathcal{N}_{s_i}} \frac{\psi_{s_i,w} \mathbf{1}(L_w = l_k)}{\hat{\pi}_{s_i}(d_w^{(\mathrm{I})} + d_w^{(\mathrm{O})})} \\
&= \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \phi(s_i) \\
&\xrightarrow{a.s.} \sum_{v \in V} \pi_v \phi(v) \\
&= C_\pi \sum_{v \in V} \sum_{w \in \mathcal{N}_v} \frac{\psi_{v,w} \mathbf{1}(L_w = l_k)}{d_w^{(\mathrm{I})} + d_w^{(\mathrm{O})}} \\
&= C_\pi \sum_{w \in V} \sum_{v \in \mathcal{N}_w} \frac{\psi_{w,v} \mathbf{1}(L_w = l_k)}{d_w^{(\mathrm{I})} + d_w^{(\mathrm{O})}} \\
&= C_\pi \sum_{w \in V} \mathbf{1}(L_w = l_k) \\
&= C_\pi |V| \theta_k.
\end{aligned}$$

The second last equation holds because $\sum_{v \in \mathcal{N}_w} \psi_{w,v} = d_w^{(\mathrm{I})} + d_w^{(\mathrm{O})}$, which is easily obtained by the definition of $\psi_{w,v}$. Similarly, we have $\lim_{n\to\infty} \frac{\breve{C}_{\mathrm{d}}}{n} \xrightarrow{a.s.} C_\pi |V|$, and then $\lim_{n\to\infty} \breve{\theta}_k^* \xrightarrow{a.s.} \theta_k$. □

Next, we propose a method for graphs such as Google scholar, Quora, and Citeseerx,[9] where we can obtain a sampled node's neighbors' in-degrees but no out-degrees. A node $w \in V$ appears as a following of $d_w^{(\mathrm{I})}$ nodes in $\mathcal{N}_w^{(\mathrm{I})}$. Therefore, node w with larger $d_w^{(\mathrm{I})} + 1$ has a larger chance to appear as a sampled node or a neighbor of sampled nodes for UNI. Based on sampled nodes and their outgoing neighbors, similar to $\breve{\theta}_k$, we estimate θ_k as

[9] http://citeseerx.ist.psu.edu/.

$$\breve{\theta}_k^{(\mathrm{O})} = \frac{1}{\breve{C}_{\mathrm{d}}^*} \sum_{i=1}^{n} \sum_{w \in \mathcal{N}_{s_i}^{(\mathrm{O})} \cup s_i} \frac{\mathbf{1}(L_w = l_k)}{\hat{\pi}_{s_i}(d_w^{(\mathrm{I})} + 1)},$$

where $\breve{C}_{\mathrm{d}}^* = \sum_{i=1}^{n} \sum_{w \in \mathcal{N}_{s_i}^{(\mathrm{O})} \cup s_i} \frac{1}{\hat{\pi}_{s_i}(d_w^{(\mathrm{I})}+1)}$.

Theorem 3 *$\breve{\theta}_k^{(O)}$ is an asymptotically unbiased estimate of θ_k.*

Proof Let $\phi(v) = \frac{\mathbf{1}(L_v = l_k)}{\hat{\pi}_v(d_v^{(\mathrm{I})}+1)} + \sum_{w \in \mathcal{N}_v^{(\mathrm{O})}} \frac{\mathbf{1}(L_w = l_k)}{\hat{\pi}_v(d_w^{(\mathrm{I})}+1)}$. Applying Lemma 1, we have

$$\begin{aligned}
&\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \frac{\mathbf{1}(L_{s_i} = l_k)}{\hat{\pi}_{s_i}(d_{s_i}^{(\mathrm{I})} + 1)} + \sum_{w \in \mathcal{N}_{s_i}^{(\mathrm{O})}} \frac{\mathbf{1}(L_w = l_k)}{\hat{\pi}_{s_i}(d_w^{(\mathrm{I})} + 1)} \\
&= \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \phi(s_i) \\
&\xrightarrow{a.s.} \sum_{v \in V} \pi_v \phi(v) \\
&= C_\pi \sum_{v \in V} \left(\frac{\mathbf{1}(L_v = l_k)}{d_v^{(\mathrm{I})} + 1} + \sum_{w \in \mathcal{N}_v^{(\mathrm{O})}} \frac{\mathbf{1}(L_w = l_k)}{d_w^{(\mathrm{I})} + 1} \right) \\
&= C_\pi \left(\sum_{v \in V} \frac{\mathbf{1}(L_v = l_k)}{d_v^{(\mathrm{I})} + 1} + \sum_{v \in V} \sum_{w \in \mathcal{N}_v^{(\mathrm{O})}} \frac{\mathbf{1}(L_w = l_k)}{d_w^{(\mathrm{I})} + 1} \right) \\
&= C_\pi \left(\sum_{v \in V} \frac{\mathbf{1}(L_v = l_k)}{d_v^{(\mathrm{I})} + 1} + \sum_{w \in V} \sum_{v \in \mathcal{N}_w^{(\mathrm{I})}} \frac{\mathbf{1}(L_w = l_k)}{d_w^{(\mathrm{I})} + 1} \right) \\
&= C_\pi \left(\sum_{v \in V} \frac{\mathbf{1}(L_v = l_k)}{d_v^{(\mathrm{I})} + 1} + \sum_{w \in V} \frac{d_w^{(\mathrm{I})} \mathbf{1}(L_w = l_k)}{d_w^{(\mathrm{I})} + 1} \right) \\
&= C_\pi \sum_{v \in V} \mathbf{1}(L_v = l_k) \\
&= C_\pi |V| \theta_k.
\end{aligned}$$

The fourth last equation holds because we have

$$\sum_{v \in V} \sum_{w \in \mathcal{N}_v^{(\mathrm{O})}} \chi(v, w) = \sum_{w \in V} \sum_{v \in \mathcal{N}_w^{(\mathrm{I})}} \chi(v, w)$$

for any function $\chi(v, w)$. Similarly, we have $\lim_{n\to\infty} \frac{\breve{C}_{\mathrm{d}}^*}{n} \xrightarrow{a.s.} C_\pi |V|$, and then $\lim_{n\to\infty} \breve{\theta}_k^{(\mathrm{O})} \xrightarrow{a.s.} \theta_k$. □

4 Edge label density estimation

Let $L_{u,v}$ denote the label of edge (u, v), with range $\{l'_1, \ldots, l'_{K'}\}$. For undirected graph G, we let $L_{u,v} = L_{v,u}$. Note that the labels of edges (u, v) and (v, u) in directed graph G_{d} may

be different. For $1 \le k \le K'$, let $0 \le \tau_k \le 1$ denote the fraction of edges with label l'_k. We define the edge label density as $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_{K'})$. In this section, we propose methods to estimate $\boldsymbol{\tau}$ for undirected and directed graphs respectively.

4.1 Simple estimators of edges densities

Ribeiro and Towsley [6] show that RW and FS sample each edge with the same probability at the steady state. Based on edges $[(s_i^-, s_i)]_{1\le i\le n}$ sampled by RW and FS, therefore, [6] estimates τ_k for undirected graph G as

$$\hat{\tau}_k = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}\left(L_{s_i^-,s_i} = l'_k\right), \quad 1 \le k \le K', \tag{5}$$

and demonstrates that $\hat{\tau}_k$ is an asymptotically unbiased estimate of τ_k. One can easily extend this estimator to compute $\boldsymbol{\tau}$ of directed graph G_{d} as

$$\begin{aligned}\hat{\tau}_k^* = \frac{1}{H_{\mathrm{d}}}\sum_{i=1}^{n} \Big(&\mathbf{1}\left(L_{s_i^-,s_i} = l'_k\right)\mathbf{1}\big((s_i^-, s_i) \in E_{\mathrm{d}}\big) \\ &+ \mathbf{1}\big(L_{s_i,s_i^-} = l'_k\big)\mathbf{1}\big((s_i, s_i^-) \in E_{\mathrm{d}}\big)\Big).\end{aligned}$$

where $H_{\mathrm{d}} = \sum_{i=1}^{n} \mathbf{1}((s_i^-, s_i) \in E_{\mathrm{d}}) + \mathbf{1}((s_i, s_i^-) \in E_{\mathrm{d}})$.

4.2 Estimators using neighborhood information of sampled nodes

In this paper, we assume that we can obtain the labels of all (resp. incoming and outgoing) edges of a node when querying the node from G (resp. G_{d}). Besides RW and FS, UNI then can also be used to sample edges. We utilize the neighborhood information of sampled nodes $[s_i]_{1\le i\le n}$ to improve the accuracy of estimating edge label density $\boldsymbol{\tau}$ for both undirected and directed graphs.

- *Undirected graph* We estimate τ_k of G as

$$\breve{\tau}_k = \frac{1}{\breve{H}}\sum_{i=1}^{n}\sum_{w\in\mathcal{N}_{s_i}} \frac{\mathbf{1}\left(L_{s_i,w} = l'_k\right)}{\hat{\pi}_{s_i}}, \quad 1 \le k \le K', \tag{6}$$

  where $\breve{H} = \sum_{i=1}^{n}\sum_{w\in\mathcal{N}_{s_i}} \hat{\pi}_{s_i}^{-1}$.

Theorem 4 *$\breve{\tau}_k$ is an asymptotically unbiased estimate of τ_k for undirected graph G.*

Proof Let $\phi(v) = \sum_{w\in\mathcal{N}_v} \frac{\mathbf{1}(L_{v,w}=l'_k)}{\hat{\pi}_v}$. Applying Lemma 1, we have

$$\begin{aligned}\lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n}\sum_{w\in\mathcal{N}_{s_i}}\frac{\mathbf{1}(L_{s_i,w}=l'_k)}{\hat{\pi}_{s_i}} &= \lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n}\phi(s_i) \\ &\xrightarrow{a.s.} \sum_{v\in V}\pi_v\phi(v) \\ &= C_\pi\sum_{v\in V}\sum_{w\in\mathcal{N}_v}\mathbf{1}(L_{v,w}=l'_k) \\ &= 2C_\pi|E|\tau_k.\end{aligned}$$

Similarly, we have $\lim_{n\to\infty} \frac{\breve{H}}{n} \to 2C_\pi|E|$, and then $\lim_{n\to\infty} \breve{\tau}_k \xrightarrow{a.s.} \tau_k$. □

- *Directed graph* We estimate τ_k of G_d as

$$\breve{\tau}_k^* = \frac{1}{\breve{H}_\mathrm{d}} \sum_{i=1}^{n} \sum_{w\in\mathcal{N}_{s_i}} \left(\frac{\mathbf{1}(L_{s_i,w} = l'_k)\mathbf{1}((s_i, w) \in E_\mathrm{d})}{\hat{\pi}_{s_i}} + \frac{\mathbf{1}(L_{w,s_i} = l'_k)\mathbf{1}((w, s_i) \in E_\mathrm{d})}{\hat{\pi}_{s_i}} \right)$$

where $\breve{H}_\mathrm{d} = \sum_{i=1}^{n} \sum_{w\in\mathcal{N}_{s_i}} \frac{\mathbf{1}((s_i,w)\in E_\mathrm{d})+\mathbf{1}((w,s_i)\in E_\mathrm{d})}{\hat{\pi}_{s_i}}$. Similar to Theorem 4, we have the following theorem.

Theorem 5 *$\breve{\tau}_k^*$ is an asymptotically unbiased estimate of τ_k for directed graph G_d.*

In summary, $\hat{\boldsymbol{\tau}} = (\hat{\tau}_1, \ldots, \hat{\tau}_{K'})$ and $\hat{\boldsymbol{\tau}}^* = (\hat{\tau}_1^*, \ldots, \hat{\tau}_{K'}^*)$ computed as described above form asymptotically unbiased estimates of $\boldsymbol{\tau}$ for undirected and directed graphs, respectively. When properties of sampled nodes' neighbors are available, we utilize all edge labels observed from the neighborhood information, and provide asymptotically unbiased estimates $\breve{\boldsymbol{\tau}} = (\breve{\tau}_1, \ldots, \breve{\tau}_{K'})$ and $\breve{\boldsymbol{\tau}}^* = (\breve{\tau}_1^*, \ldots, \breve{\tau}_{K'}^*)$ of $\boldsymbol{\tau}$ for undirected and directed graphs, respectively.

5 High degree node detection

In this section, we study the problem of detecting nodes with the largest degrees in undirected graph $G = (V, E)$. Let S be the set of nodes sampled by methods such as RW. Previous methods use the high degree nodes in S to estimate high degree nodes in the original graph G [18,19]. In [19], weighted RW (WRW) is used to detect high degree nodes. WRW can be viewed as a RW over a weighted graph, where each edge $(u, v) \in E$ has a positive weight $w_{u,v} = w_{v,u}$ [20]. At each step, WRW selects the next-hop node v randomly from the neighbors of the current node u with probability proportional to weight $w_{u,v}$, which can be achieved with computational complexity $O(d_u)$. WRW (with well defined edge weights) and RW are fast for detecting high degree nodes, since they are biased to sample high degree nodes [6,19]. Note that the WRW proposed in [19] sets weight $w_{u,v} = (d_u d_v)^\beta$, which indicates that at each step their WRW requires to obtain the degrees of current sampled node's neighbors. However, their description does not account for the cost of retrieving this information. In [21], a method, expansion sampling (XS), is proposed for detecting high degree nodes. Denote by $\mathcal{N}_S$ the neighborhood of S, where $\mathcal{N}_S$ consists of nodes in $V - S$ that are neighbors of nodes in S, that is

$$\mathcal{N}_S = \{u : \exists v \in S, (u, v) \in E \wedge u \in V - S\}.$$

Starting from a random node s, and $S = \{s\}$, XS adds the node in $\mathcal{N}_S$ that has the most neighbors in $V - (\mathcal{N}_S \cup S)$ to S, which can be achieved with computational complexity $O(|\mathcal{N}_S|)$, and repeats this process. For a node $u \in \mathcal{N}_S$, denote by d_u^S the number of edges between u and nodes in S, and $d_u^{\mathcal{N}_S}$ the number of edges between u and nodes in $\mathcal{N}_S$. Then, the number of its neighbors in $V - (\mathcal{N}_S \cup S)$ equals $d_u - d_u^S - d_u^{\mathcal{N}_S}$. One can compute d_u and d_u^S solely based on the knowledge of edges of nodes in S. However, $d_u^{\mathcal{N}_S}$ cannot be obtained based on the available information of S and $\mathcal{N}_S$. In order to identify the node in $\mathcal{N}_S$ that has the most neighbors in $V - (\mathcal{N}_S \cup S)$, it is necessary to crawl all nodes in $\mathcal{N}_S$. The original description of XS [21] does not account for this cost. To solve this problem, [22] develops a method named SEC to select the node $v \in \mathcal{N}_S$ with the most neighbors in S as the next node to crawl. When sampled nodes' degrees are given without extra crawling cost, i.e., each node

Table 2 Computational, memory, and crawling cost of obtaining n sampled nodes for detecting high degree nodes

Method	Computational cost	Memory cost	Crawling cost
Without free neighborhood information			
RW	$O(n)$	$O(n)$	$O(n)$
WRW	$O(\bar{d}n)$	$O(n)$	$O(\bar{d}n)$
XS	$O(\bar{d}^2n^2)$	$O(\bar{d}n)$	$O(\bar{d}n)$
SEC	$O(\bar{d}^2n^2)$	$O(\bar{d}n)$	$O(n)$
With free neighborhood information			
RW	$O(n)$	$O(n)$	$O(n)$
WRW	$O(\bar{d}n)$	$O(n)$	$O(n)$
MXS	$O(\bar{d}^2n^2)$	$O(\bar{d}n)$	$O(n)$

The crawling cost refers to the number of queries a crawler sent to the network of interest

has the knowledge of its neighbors' degrees, we can simply extend RW, SEC, and WRW to utilize this information. That is, we use high degree nodes in $S \cup \mathcal{N}_S$ to estimate high degree nodes in G. In addition, we propose a more crawling efficient method *Modified XS* (MXS). It requires no extra crawling cost to identify the node in $\mathcal{N}_S$ that has the most neighbors in $V - S$, i.e., the node $u \in \mathcal{N}_S$ with the largest value of $d_u - d_u^S$, because d_u^S can be easily computed based on the knowledge of edges from nodes in S to u. Thus, we add this node u to S at each step and repeats this process. Finally, we output high degree nodes in $\mathcal{N}_S \cup S$ as the final result.

The above method can be easily modified to identify nodes with the largest in- or out-degrees for directed graphs such as Sina microblog and Xiami, where a node has the knowledge of its neighbors' out- and in-degrees. Here, at each step, MXS adds the node $w \in \mathcal{N}_S$ with the largest $d_w^{(\mathrm{I})} + d_w^{(\mathrm{O})}$ to S. Table 2 shows the computational, memory, and crawling cost of obtaining n sampled nodes (i.e., repeating the sampling process n times) for all above methods. We can see that RW is the most computational and memory efficient method. In this paper, however, our aim is to accurately detect high degree nodes by using a small crawling cost, i.e., the number of queries posted to the network of interest. In our later experiments, we demonstrate that our method MXS is much more accurate than the other methods under the same crawling cost.

6 Shortest path discovery

In this section, we study the problem of performing topology discovery and message routing with incomplete topological information, which is important for applications such as the discovery of shortest paths between OSN users and routing algorithms (e.g. Bubble Rap [23]) for delivering messages between users using an OSN. Formally, the problem is: Two nodes u and v are looking for the shortest path on undirected graph G. Ribeiro et al. [24] find that a RW has the ability to observe a large fraction of edges by visiting a relatively small number of nodes on power law graphs. Here an edge is observed when at least one of its endpoints is visited by the RW. They propose a RW-based shortest path discovery algorithm that works as follows: Two RWs are started from u and v separately. Each RW takes n steps. Let S be the set of nodes sampled by two RWs. They use the shortest path in observed graph $G^* = (V^*, E^*)$ for routing between u and v, where $V^* = S \cup \mathcal{N}_S$ and E^* consists of edges in E which have at least one endpoint in S. From Sect. 5 and our experimental results in Sect. 7, we know that

Table 3 Overview of graph datasets used in our experiments

Graph	LCC		
	Nodes	Edges	Directed-edges
Xiami [25]	1,748,010	16,015,779	16,568,449
YouTube [26]	1,134,890	2,987,624	4,942,035
Flickr [26]	1,624,992	15,476,835	22,477,014
Soc-Epinions [27]	75,877	405,739	811,478
Soc-Slashdot [28]	77,360	469,180	828,161

"Directed-edges" refers to the number of directed edges in a directed graph, "edges" refers to the number of edges in an undirected graph obtained by ignoring the direction of edges, and "LCC" refers to the largest connected component of a given graph

MXS can effectively find high degree nodes and observe more edges based on neighborhood information. Thus, we extend MXS to accelerate the performance of shortest path discovery. That is, we perform an MXS starting from u and v, respectively, and use the shortest path in graph $G^* = (V^*, E^*)$ observed by MXS for routing between u and v.

7 Experiments

7.1 Datasets

We perform our experiments on a variety of real-world networks that are summarized in Table 3. Xiami is a popular website devoted to music streaming and recommendations. Similar to Twitter, Xiami builds a social network based on follower and following relationships. Flickr and YouTube are popular photo sharing and video sharing websites. In these websites, a user can subscribe to other user updates such as blogs and photos. These networks can be represented by direct graphs, with nodes representing users and a directed edge from u to v represents that user u subscribes to user v. Epinions is a who-trusts-whom OSN providing general consumer reviews, where a directed edge from u to v represents that user u trusts user v. Slashdot is a technology-related news website for its specific user community, where a directed edge from u to v represents that user u tags user v as a friend or foe. In the following experiments, we evaluate our methods in comparison with previous methods based on the largest connected component (LCC) of these graphs under the same sampling budget n, where n is defined as the number of sampled nodes.

7.2 Results of node label density estimation

Error metric In our study, besides θ_k (i.e., the fraction of nodes with label l_k), we also estimate $\xi_k = \sum_{i=k+1}^{K} \theta_i$, the complementary cumulative distribution function (CCDF) of $\boldsymbol{\theta}$, which is the statistic of choice when it comes to display degree distributions. For estimator $\hat{\theta}_k$, we define the normalized root mean square error (NMSE) as $\mathrm{NMSE}(\hat{\theta}_k) = \frac{\sqrt{E[(\hat{\theta}_k - \theta_k)^2]}}{\theta_k}$, $k = 1, 2, \ldots$. In the following experiments, we use 1000 independent runs to estimate $E[(\hat{\theta}_k - \theta_k)^2]$. Similarly, we define the NMSE of the CCDF of $\boldsymbol{\theta}$, which we denote as the CNMSE to avoid confusion with the NMSE of $\boldsymbol{\theta}$.

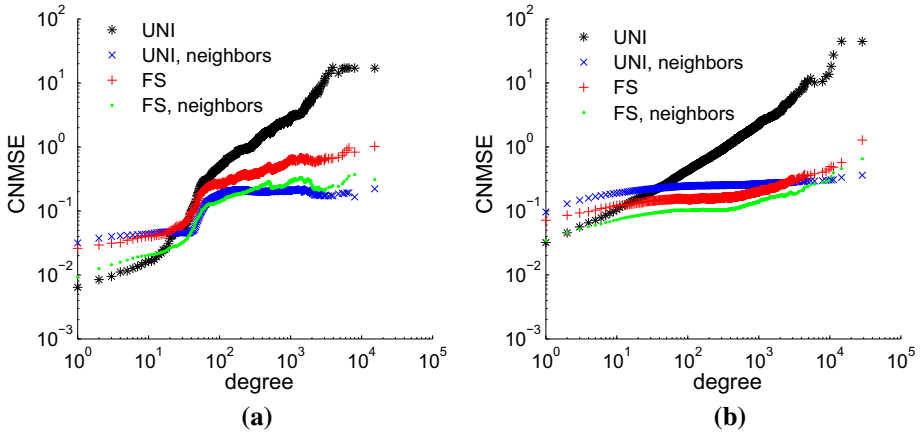


Fig. 2 Results of degree distribution estimations for undirected graphs, $n = 0.001|V|$. **a** Xiami, **b** YouTube

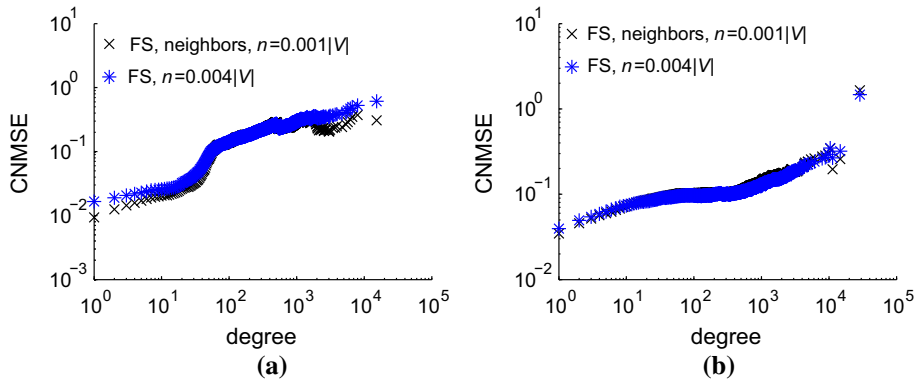


Fig. 3 To achieve the same error, the regular FS requires at least 4× the number of the samples of FS with neighborhood information. **a** Xiami, **b** YouTube

Results of characterizing undirected graphs Figure 2 shows the CNMSEs of estimates of degree distribution, where we set sampling budget $n = 0.001|V|$. For crawling methods, we only study FS and set its parameter $m = 10$. The result of RW is similar to that of FS. Figure 2 shows that degree distribution estimates produced by UNI and FS using neighborhood information almost have the same accuracy. For FS, the degree distribution estimate greatly improves when neighborhood information is used, which is almost *twice* as accurate than the regular FS without using neighborhood information for Xiami. [6,29] show that NMSEs are roughly proportional to $\frac{1}{\sqrt{n}}$. It indicates that FS using neighborhood information requires *4 times* faster than the regular FS method to achieve the same accuracy, which is consistent with our results shown in Fig. 3. For UNI method, the degree distribution estimator based on using the neighborhood information of sampled nodes exhibits larger errors than the estimator given by sampled nodes for small degrees (degrees smaller than 20 and 30 for Xiami and YouTube respectively). For the degree distribution estimator given by neighbors of sampled nodes, we can see that FS using neighborhood information is more accurate than the other methods for most degrees. We also evaluate the performance of FS using neighborhood information for estimating the location distribution of users in Xiami. Figure 4 shows

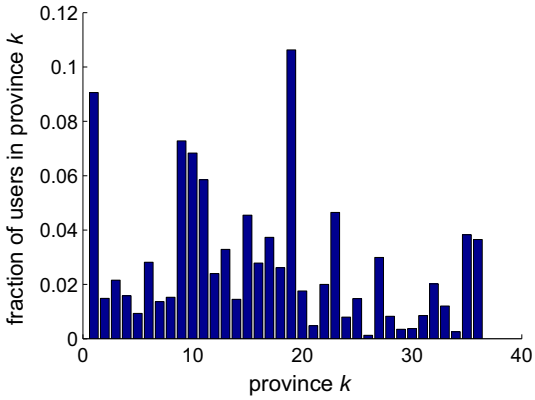


Fig. 4 The location distribution of users in Xiami

Table 4 (Xiami) Province numbers and corresponding names

1. Beijing	2. Tianjin	3. Hebei	4. Shanxi	5. Inner Mongolia
6. Liaoning	7. Jilin	8. Heilongjiang	9. Shanghai	10. Jiangsu
11. Zhejiang	12. Anhui	13. Fujian	14. Jiangxi	15. Shandong
16. Henan	17. Hubei	18. Hunan	19. Guangdong	20. Guangxi
21. Hainan	22. Chongqing	23. Sichuan	24. Guizhou	25. Yunnan
26. Tibet	27. Shannxi	28. Gansu	29. Qinghai	30. Ningxia
31. Xinjiang	32. Taiwan	33. Hong Kong	34. Macao	35. Null
36. Overseas				

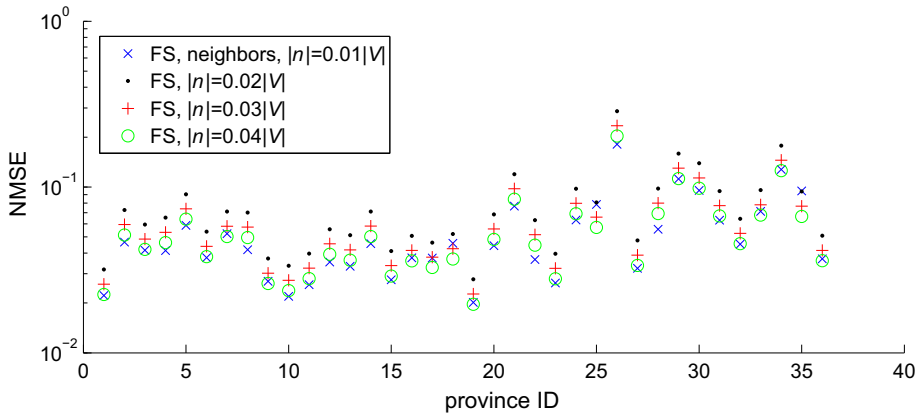


Fig. 5 To achieve the same error, regular FS requires at least 2 to 4× the number of the samples of FS with neighborhood information

the real value, where the province numbers and corresponding names are shown in Table 4. Figure 5 shows the NMSEs of FS using neighborhood information in comparison with the regular FS method. We can see that FS using neighborhood information is *4 times* faster than the regular FS method for most provinces and is *2–3 times* faster than the regular FS method for the other provinces.

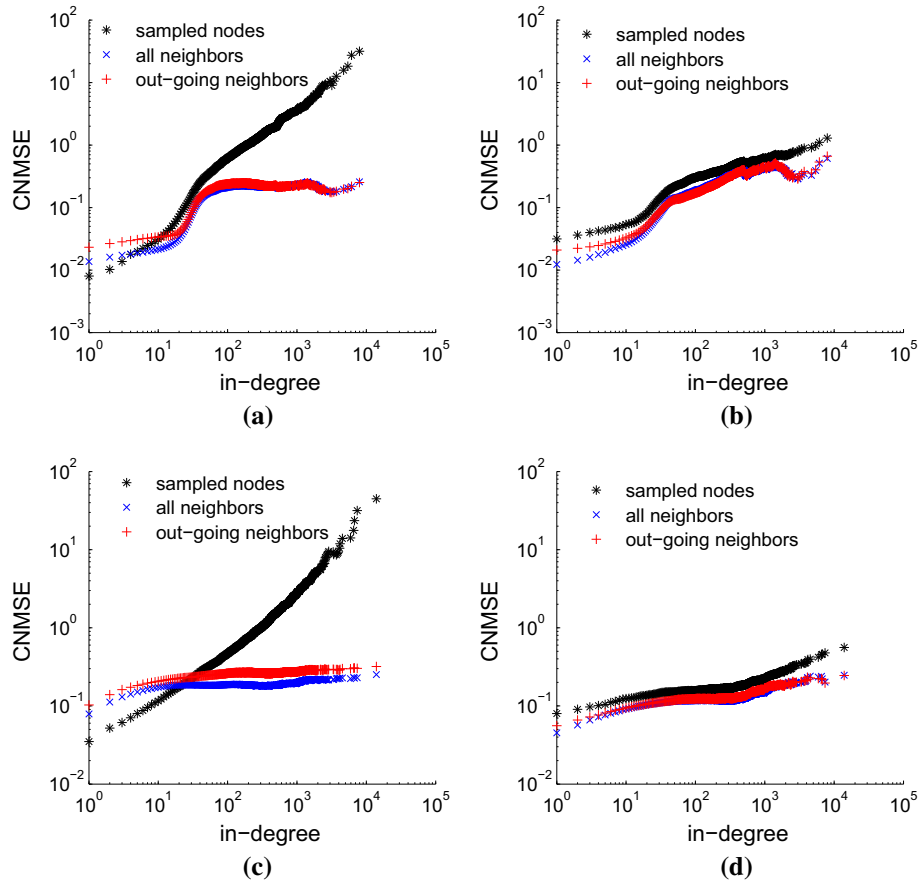


Fig. 6 Results of in-degree distribution estimations for directed graphs, $n = 0.001|V|$. **a** Xiami, UNI; **b** Xiami, FS; **c** YouTube, UNI; **d** YouTube, FS

Results of characterizing directed graphs For directed graphs, Fig. 6 shows results for in-degree distribution estimates. When in-degrees and out-degrees of sampled nodes are available, the in-degree distribution estimator given by neighbors of nodes sampled by FS *outperforms* the estimator given by the sampled nodes. For small in-degrees (3 and 18 for Xiami and YouTube respectively), the in-degree distribution estimator given by neighbors of nodes sampled by UNI exhibits larger errors than the estimator given by the sampled nodes. Meanwhile, the results show that we can also give an accurate in-degree distribution estimate given by outgoing neighbors of sampled nodes, which is a little less accurate than the estimate obtained by all neighbors' information. Figure 7 shows the results of the mixture estimator in (4). We observe that the mixture estimator *outperforms* the estimator based on sampled nodes and the estimator based on neighbors of sampled nodes.

Results in comparison with state-of-the-art methods Let c denote the cost of UNI, i.e., the average number of IDs queried until one valid ID is obtained. For example, Flickr has a random node sampling cost of $c = 77$ [29]. Here, we set the cost of crawling methods FS and RW as 1. Next we compare with performance of crawling methods with social sampling (SS), a node sampling method proposed by Dasgupta et al. [5]. Here, SS is equivalent to

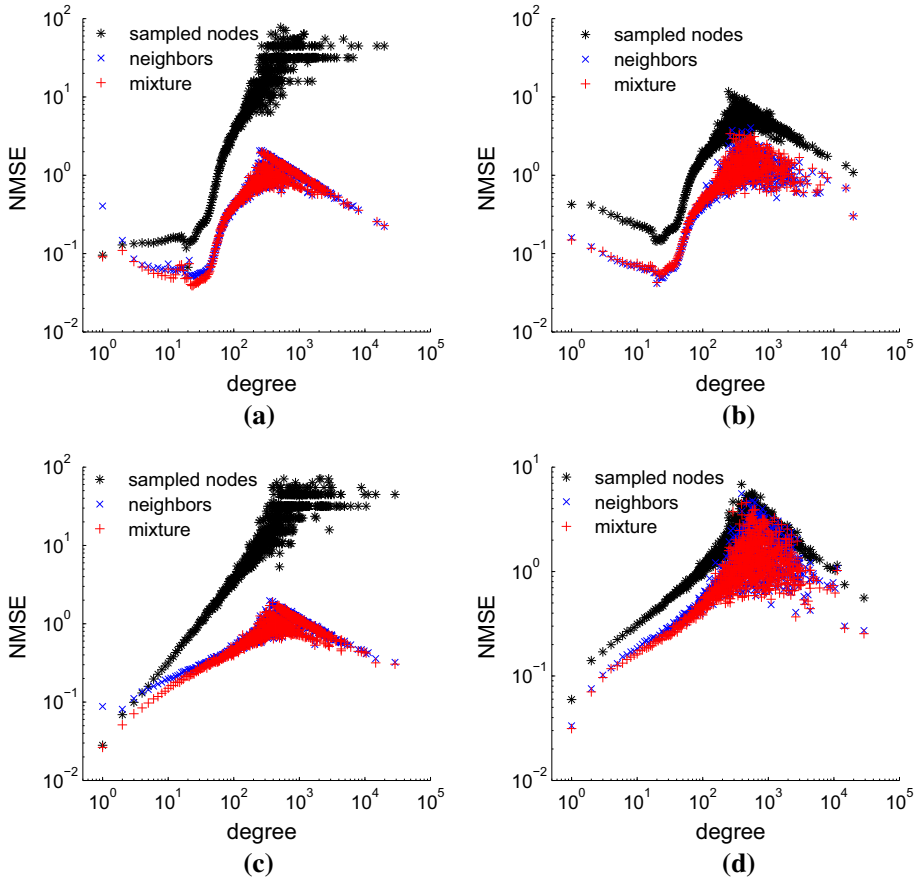


Fig. 7 Results of degree distribution estimations for the mixture estimator, $n = 0.001|V|$. **a** Xiami, UNI; **b** Xiami, FS; **c** YouTube, UNI; **d** YouTube, FS

the estimator given by neighbors of nodes sampled by UNI. Figure 8 shows that SS exhibits larger errors as c increases. When sampling cost $c = 10$, FS and RW using neighborhood information are much more accurate than SS under the same sampling budget. Meanwhile, we can see that FS using neighborhood information exhibits smaller errors than RW using neighborhood information.

7.3 Results of edge label density estimation

Results of estimating the joint degree distribution We evaluate the performance of our method for estimating the joint degree distribution $\boldsymbol{\tau} = [\tau_{i,j}]_{0<j\leq i}$ for undirected graph G, where $\tau_{i,j}$ is the fraction of edges consisting of two nodes with degrees i and j separately. For two-dimensional distribution $\boldsymbol{\tau}$, we define δ as

$$\delta = \sqrt{\sum_{0<j\leq i} (\hat{\tau}_{i,j} - \tau_{i,j})^2},$$

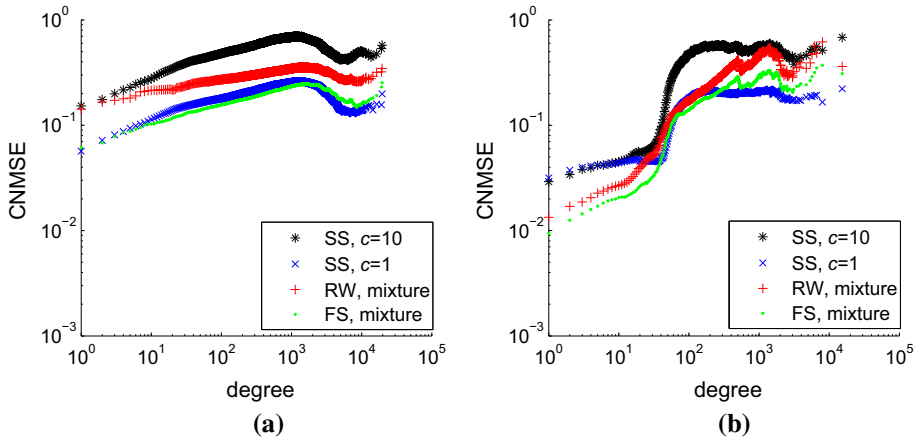


Fig. 8 Results of degree distribution estimations for different node sampling cost c, $n = 0.001|V|$. **a** Flickr, **b** Xiami

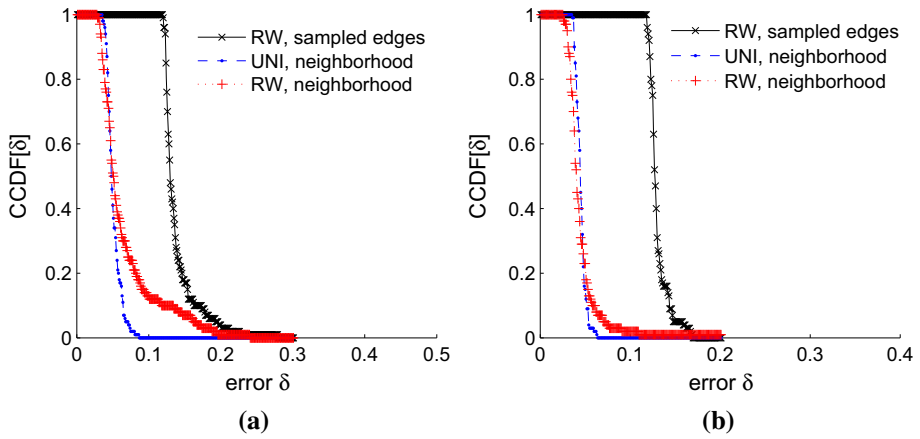


Fig. 9 CCDFs of errors of joint degree distribution estimates, $n = 0.001|V|$. **a** Soc-Epinions, **b** Soc-Slashdot

which is a metric that measures the error of its estimate $\hat{\boldsymbol{\tau}}$. Figure 9 shows the CCDF of δ for 1,000 independent estimates, where the sampling budget is $n = 0.001|V|$. It shows that RW and UNI using sampled nodes' neighborhood information are more accurate. All estimates have errors larger than 0.1 when we have no knowledge of sampled nodes' degrees. More than 85% of estimates have errors smaller than 0.1 when sampled nodes' degrees are available.

Results of estimating the edge label density Let us illustrate how to apply the edge label density estimation. Consider the directed graph of Xiami, 53.8% of users are male (M), 37.5% are female (F), and 8.7% are unknown (U). A directed edge (u, v) is classified into the following 9 types when the edge label is defined as $u.gender \rightarrow v.gender$: (1) M→M, (2) M→F, (3) M→U, (4) F→M, (5) F→F, (6) F→U, (7) U→M, (8) U→F, (9) U→U. Figure 10 shows the edge label density $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_9)$, where τ_i $(1 \leq i \leq 9)$ is the fraction of type i edges. Figure 11 shows the result of estimating $\boldsymbol{\tau}$. Similarly, we find that RW and UNI using

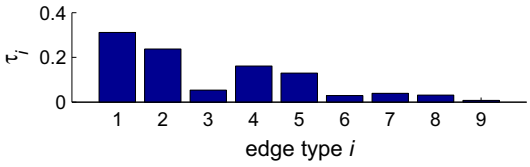


Fig. 10 (Xiami) Density of edges with different types. Type 1: M→M, 2: M→F, 3: M→U, 4: F→M, 5: F→F, 6: F→U, 7: U→M, 8: U→F, 9: U→U

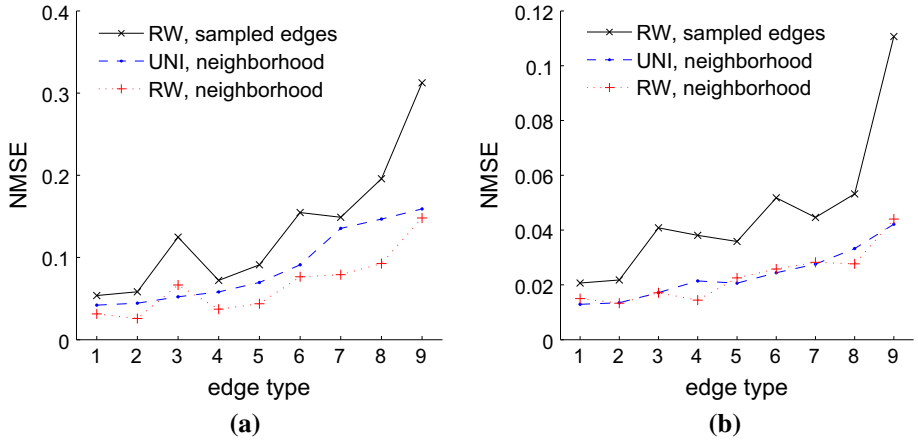


Fig. 11 (Xiami, gender-gender) NMSEs of edge label density estimates. **a** $n = 0.001|V|$, **b** $n = 0.01|V|$

sampled nodes' neighborhood information exhibit smaller errors, and are two times more accurate than the regular RW method. Here, we omit the result of FS, which is similar to that of RW.

7.4 Results of high degree node detection

In this section, we study the performance of our method MXS for detecting top high degree nodes in comparison with state-of-the-art methods. Similar to node and edge label density estimation, we evaluate all methods under the same crawling cost, i.e., the same sampling budget n. Figure 12 shows the results of previous methods for detecting top-100 high degree nodes, where the edge weight function is defined as $w_{u,v} = (d_u d_v)^\beta$ for WRW. For previous methods without the free neighborhood information of sampled nodes, we assume that XS and WRW both must crawl sampled nodes' neighbors to obtain their degrees. Figure 12 shows that RW, SEC, WRW, and XS need to sample more than $n = 0.01|V|$ nodes to obtain an accurate result (about 90%) of top-100 high degree nodes. To detect top-100 high degree nodes using the free neighborhood information of sampled nodes, Fig. 13 shows that on average the simple extensions of RW, SEC, and WRW reduce the crawling cost to about $n = 0.0001|V|$ to obtain about 90% of top-100 high degree nodes, and our method MXS is the most efficient one, which further reduces the crawling cost to $n = 0.00001|V|$. A total of 1,000 runs are used to produce the averages seen in the figure. For detecting top-100 high out- and in-degree nodes of directed graphs, similarly, Figs. 14 and 15 show that our method MXS is 10 times more crawling efficient than the other simple extensions of WRW and RW using the free neighborhood information of sampled nodes. Here, the edge weight function of WRW is defined as $w_{u,v} = (d_u^{(\mathrm{O})} + d_u^{(\mathrm{I})})^\beta (d_v^{(\mathrm{O})} + d_v^{(\mathrm{I})})^\beta$.

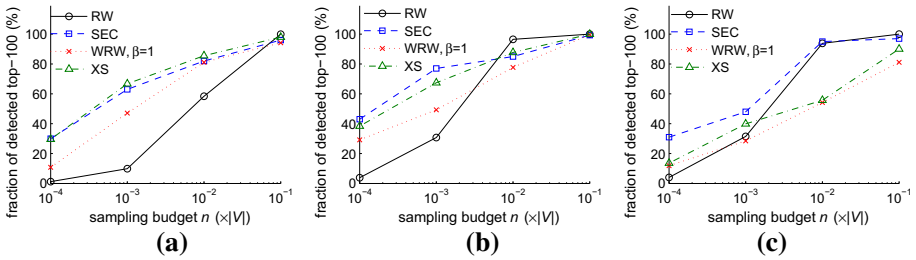


Fig. 12 (Previous methods) Results of top-100 high degree node detection. **a** Xiami, **b** Flickr, **c** YouTube

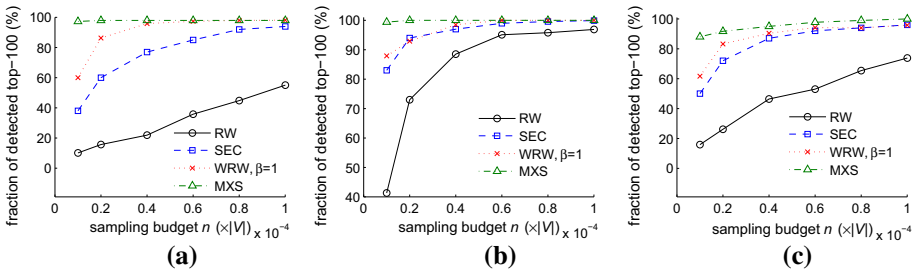


Fig. 13 (Our methods, using neighborhood information of sampled nodes) Results of top-100 high degree node detection. **a** Xiami, **b** Flickr, **c** YouTube

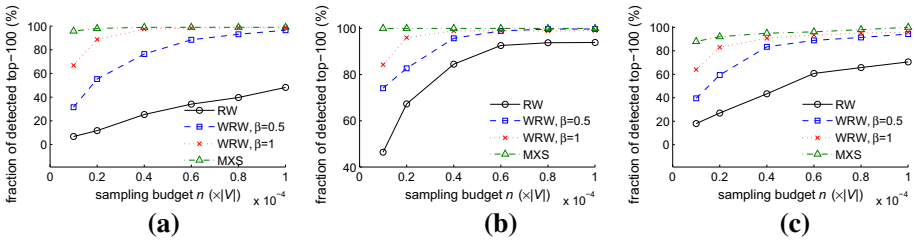


Fig. 14 (Our methods, using neighborhood information of sampled nodes) Results of top-100 high out-degree node detection. **a** Xiami, **b** Flickr, **c** YouTube

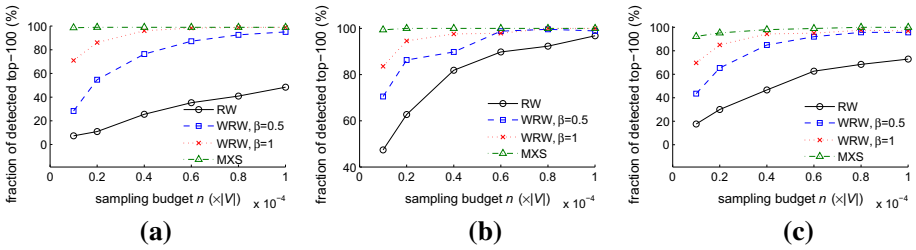


Fig. 15 (Our methods, using neighborhood information of sampled nodes) Results of top-100 high in-degree node detection. **a** Xiami, **b** Flickr, **c** YouTube

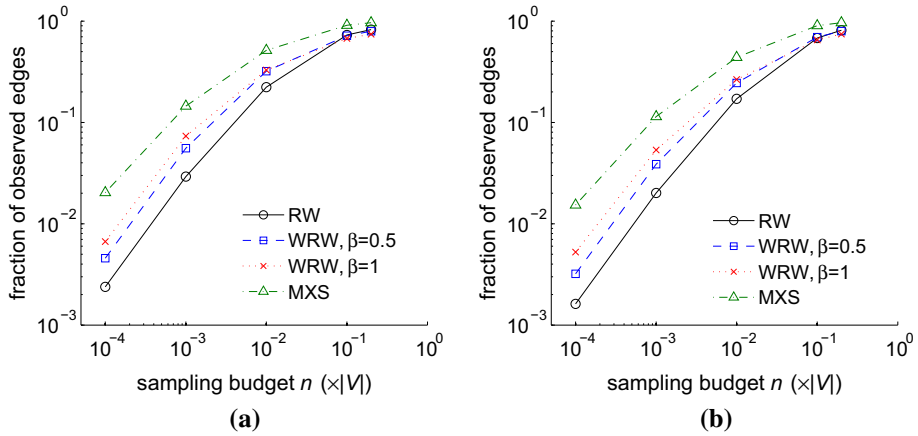


Fig. 16 Fractions of observed edges. **a** Soc-Epinions, **b** Soc-Slashdot

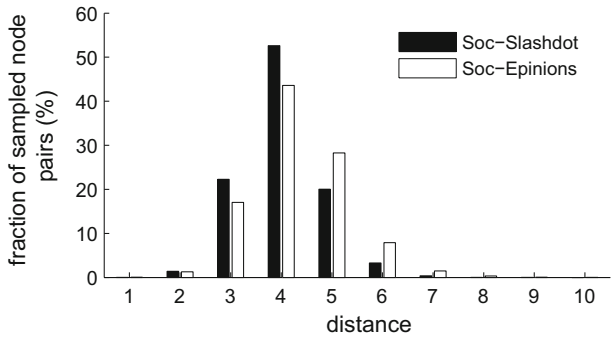


Fig. 17 (Soc-Slashdot and Soc-Epinions) Fractions of sampled node pairs with a given distance

7.5 Results of shortest path discovery

Figure 16 shows that MXS observes significantly more edges than WRW and RW under the same sampling budget n, where the edge weight function is defined as $w_{u,v} = (d_u d_v)^\beta$ for WRW. Next, we evaluate our MXS based shortest path discovery method in comparison with the regular WRW- and RW-based methods in [24]. We use two metrics to evaluate the performance of detecting the shortest paths of 10,000 node pairs: (1) the ratio of shortest path discovery failures. For two nodes with distance $d < \infty$ in G, let d^* be the length of the shortest path observed by sampling methods. When there exists no path observed for two nodes of interest, we denote $d^* = \infty$, and a failure is reported; (2) For all $d^* < \infty$, we also use the average value of $d^* - d$ as a metric to measure the accuracy of detecting the shortest paths. Figures 18 and 19 show results for 10,000 node pairs generated randomly, where the sampling budget is set as $n = 20$. Figure 17 shows the fraction of sampled node pairs with given distances (the length of the shortest paths in original graphs) for Soc-Slashdot and Soc-Epinions. Figure 18 shows the fraction of failures as a function of the distance. Y axis shows the fraction of failures for node pairs with a given distance. We can see that RW and WRW generate a large fraction of failures especially for node pairs with a long distance, e.g., more than 20% of failures for node pairs with distance larger than 6 for Soc-Epinions.

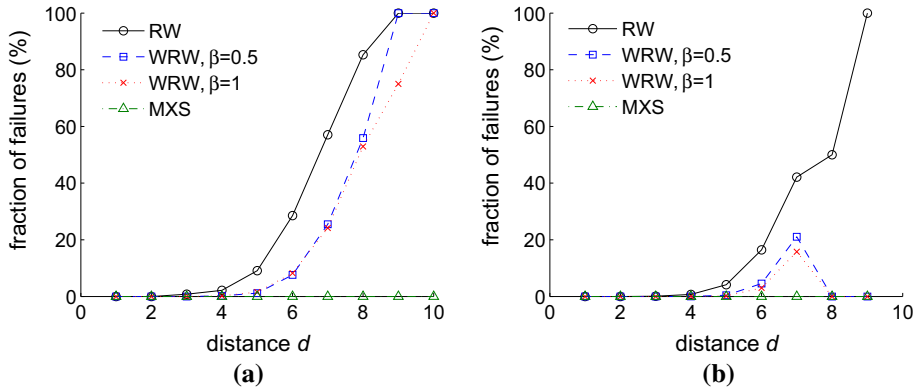


Fig. 18 Performance of shortest path discovery, $n = 20$. **a** Soc-Epinions, **b** Soc-Slashdot

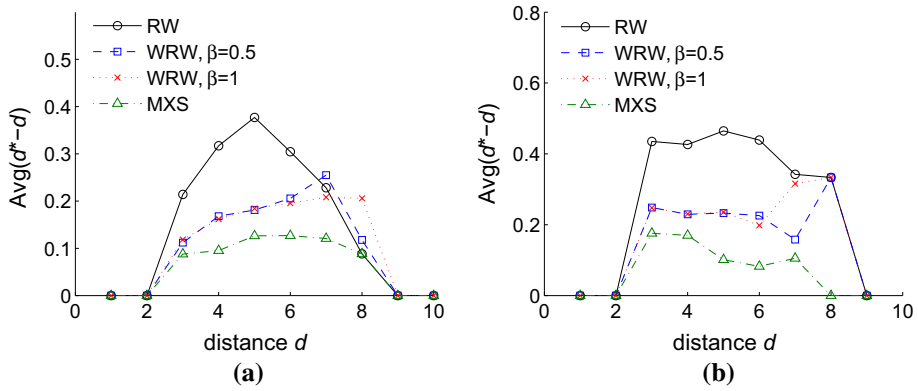


Fig. 19 NMSEs of detected shortest path lengthes, $n = 20$. **a** Soc-Epinions, **b** Soc-Slashdot

However, our method MXS almost has no failure. Moreover, Fig. 19 shows that MXS usually discovers shorter paths in comparison with RW and WRW. On average, the $d^* - d$ of MXS is 2 and 4 times smaller than WRW and RW respectively.

8 Related work

Graph sampling methods without using free neighborhood information Maiya and Berger-Wolf [3] empirically investigate the performance of a number of subgraph sampling methods (e.g., breadth-first search, RWs, etc.) and their performance in respect to various topological properties (e.g., degree and clustering coefficient). The literature also shows a variety of subgraph sampling works without convergence or accuracy guarantees [1,2], which have been empirically tested over a variety of networks. The above works [1–3] also consider subgraph sampling techniques that can preserve other metrics, such as the eigenvalues of the original network [1], but without accuracy guarantees. Breadth-First-Search (BFS) introduces a large bias toward high degree nodes, and it is difficult to remove this bias in general, although it can be ameliorated if the network in question is almost random [31]. RW is biased to sample high degree nodes; however, its bias is known and can be easily corrected [6]. RW in

Table 5 Our methods in comparison with state-of-the-art methods

	Free neighborhood information	
	Not using	Using
Node label density estimation	[6,7,30,32–34]	[5] requires expensive UNI APIs Our method is a crawling based method
Edge label density estimation	[6]	Our method
High degree node detection	[18,19,21,22]	Our method
Shortest path discovery	[24]	Our method

the form of respondent driven sampling (RDS) [32,33] has been used to estimate population densities using snowball samples of sociological studies. RDS was developed for small social networks with hidden links, while our method considers large OSNs without hidden links. The Metropolis-Hasting RW (MHRW) [34] modifies the RW procedure, aimed at sampling nodes with equal probability. However, [8] proves that MHRW degree distribution estimates perform poorly in comparison with RWs, more markedly for large degree nodes whose error grows proportionally to the degree value. Empirically, the accuracy of RW and MHRW has been compared in [7,35] and experimental results demonstrate that RW is consistently more accurate than MHRW. In addition to node label density estimation, a considerable attention has been given to develop crawling methods to detect high degree nodes [18,19,21,22] and uncover shortest paths [24].

Graph sampling methods using free neighborhood information Few network sampling methods use neighborhood information to provide accurate estimates that have convergence guarantees. The work closest to ours is Dasgupta et al. [5]. Dasgupta et al. [5] randomly sample nodes (either uniformly or with a known bias) and then use neighborhood information to improve its unbiased estimator. However, randomly sampling nodes is practical only if performed uniformly (in our scenarios, rejection sampling to bias the samples makes little sense) and suffers from the low query rate in NoSQL graph databases and Web APIs. Dasgupta et al. [5] partially compensate the low query rate through the use of neighborhood information present in the node query reply of a number of major OSNs. Moreover, their estimators require the knowledge of sampled nodes' neighbors' degrees, which incurs extra query costs when applied to OSNs such as Quora and Sina microblog that provide neighbors' in- and out-degrees but no degrees in the node query reply. Kurant et al. [30] design a RW-based method that uses a weighted RW to perform stratified sampling on OSNs. These weights are computed using neighborhood information. They use their technique on Facebook and show that their stratified sampling technique achieves higher estimation accuracy than other methods. However, the neighborhood information in their method is limited to helping find random walk weights and not used in the estimator. Interestingly, our estimator can be easily combined with the weighted random walk in [30] to improve its accuracy. Table 5 summarizes the contributions of our work in comparison with state-of-the-art methods.

9 Conclusions

In this paper, we study the problem of estimating characteristics for graphs where nodes have knowledge of their neighbors' properties. This feature is actually quite common in networks, such as Facebook, Google scholar, Sina microblog, and Citeseerx. To utilize this

extra neighborhood information, we develop novel estimators of node and edge label densities from sampling which have provable convergence and accuracy guarantees. Our experimental results show that our estimators drastically reduce (by 4-fold) the number of samples required to achieve the same estimation accuracy. We also adapt known techniques to detect high degree nodes and the shortest paths between a subset of nodes, and our experimental results demonstrate that our methods are two orders of magnitude faster than state-of-the-art methods.

Acknowledgements The authors wish to thank the anonymous reviewers for their helpful feedback. This work was supported in part by Army Research Office Contract W911NF-12-1-0385, and ARL under Cooperative Agreement W911NF-09-2-0053. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied of the ARL, or the U.S. Government. The work was also supported in part by National Natural Science Foundation of China (61603290, 61602371, U1301254), Ministry of Education & China Mobile Research Fund (MCM20160311), China Postdoctoral Science Foundation (2015M582663), Natural Science Basic Research Plan in Zhejiang Province of China (LGG18F020016), Natural Science Basic Research Plan in Shaanxi Province of China (2016JQ6034, 2017JM6095), Shenzhen Basic Research Grant (JCYJ20160229195940462).

References

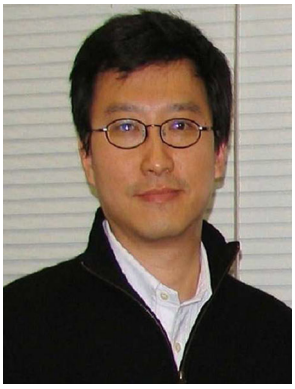
1. Leskovec J, Faloutsos C (2006) Sampling from large graphs. In: SIGKDD, pp 631–636
2. Hubler C et al (2008) Metropolis algorithms for representative subgraph sampling. In: ICDM, pp 283–292
3. Maiya AS, Berger-Wolf TY (2011) Benefits of bias: towards better characterization of network sampling. In: SIGKDD, pp 105–113
4. Ahmed NK et al (2012) Network sampling: from static to streaming graphs. TKDD 8(2):7:1–7:56
5. Dasgupta A et al (2012) Social sampling. In: SIGKDD, pp 235–243
6. Ribeiro B, Towsley D (2010) Estimating and sampling graphs with multidimensional random walks. In: IMC, pp 390–403
7. Gjoka M et al (2010) Walking in Facebook: a case study of unbiased sampling of OSNs. In: INFOCOM, pp 2498–2506
8. Ribeiro B, Towsley D (2012) On the estimation accuracy of degree distributions from graph sampling. In: CDC, pp 1–6
9. Avrachenkov K et al (2010) Improving random walk estimation accuracy with uniform restarts. In: WAW, pp 98–109
10. Graybill FA, Deal RB (1959) Combining unbiased estimators. Biometrics 15(4):543–550
11. Lovász L (1993) Random walks on graphs: a survey. Combinatorics 2:1–46
12. Ribeiro B et al (2010) Multiple random walks to uncover short paths in power law networks. In: INFOCOM NetSciCom, pp 1–6
13. Roberts GO, Rosenthal JS (2004) General state space Markov chains and MCMC algorithms. Probab Surv 1:20–71
14. Jones GL (2004) On the Markov chain central limit theorem. Probab Surv 1:299–320
15. Kurant M et al (2011) Walking on a graph with a magnifying glass: stratified sampling via weighted random walks. In: SIGMETRICS, pp 281–292
16. Horvitz DG, Thompson DJ (1952) A generalization of sampling without replacement from a finite universe. JASA 47:663–685
17. Lee CH et al (2012) Beyond random walk and Metropolis–Hastings samplers: Why you should not backtrack for unbiased graph sampling. In: SIGMETRICS/Performance, pp 319–330
18. Lim Y et al (2011) Online estimating the k central nodes of a network. In: NSW, pp 1–6
19. Cooper C et al (2012) A fast algorithm to find all high degree vertices in power law graphs. In: WWW LSNA, pp 1007–1016
20. Coppersmith D et al (1993) Random walks on weighted graphs, and applications to on-line algorithms (extended). J ACM 40:421–453
21. Maiya AS, Berger-Wolf TY (2010) Online sampling of high centrality individuals in social networks. In: PAKDD, pp 91–98
22. Maiya AS, Berger-Wolf TY (2011) Benefits of bias: towards better characterization of network sampling. In: SIGKDD, pp 105–113

23. Hui P et al (2008) BUBBLE Rap: social-based forwarding in delay tolerant networks. In: MobiHoc, pp 241–250
24. Ribeiro B et al (2012) Multiple random walks to uncover short paths in power law networks. In: Infocom NetSciCom, pp 1–6
25. Wang P et al (2012) Sampling contents distributed over graphs. Technical Report TR-1201, Xi'an Jiaotong University
26. Mislove A et al (2007) Measurement and analysis of online social networks. In: IMC, pp 29–42
27. Richardson M et al (2003) Trust management for the semantic web. In: ISWC, pp 351–368
28. Leskovec J et al (2009) Community structure in large networks: natural cluster sizes and the absence of large well-defined clusters. Internet Math 6(1):29–123
29. Ribeiro B et al (2012) Sampling directed graphs with random walks. In: INFOCOM, pp 1692–1700
30. Kurant M et al (2011) Walking on a graph with a magnifying glass: stratified sampling via weighted random walks. In: SIGMETRICS, pp 241–252
31. Kurant M et al (2011) Towards unbiased BFS sampling. JSAC 29(9):1799–1809
32. Heckathorn DD (2002) Respondent-driven sampling II: deriving valid population estimates from chain-referral samples of hidden populations. Soc Probl 49(1):11–34
33. Salganik MJ, Heckathorn DD (2004) Sampling and estimation in hidden populations using respondent-driven sampling. Sociol Methodol 49(1):11–34
34. Stutzbach D et al (2009) On unbiased sampling for unstructured peer-to-peer networks. TON 17(2):377–390
35. Rasti AH et al (2009) Respondent-driven sampling for characterizing unstructured overlays. In: INFOCOM Mini-conference, pp 2701–2705

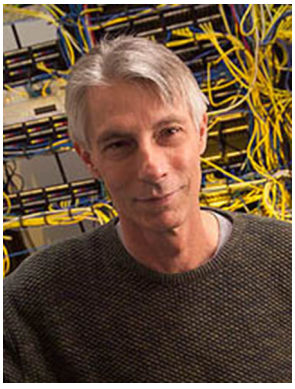
Pinghui Wang received the B.S. degree in information engineering and the Ph.D. degree in automatic control from Xi'an Jiaotong University, Xi'an, China, in 2006 and 2012 respectively. He is currently an associate professor in MOE Key Laboratory for Intelligent Networks and Network Security at Xi'an Jiaotong University. His research interests include Internet traffic measurement and modeling, abnormal detection, and online social network measurement.

Junzhou Zhao received the B.S. degree in information engineering and the Ph.D. degree in automatic control from Xi'an Jiaotong University, Xi'an, China, in 2008 and 2015 respectively. He is currently a Postdoc Fellow in the Division of Computer, Electrical and Mathematical Sciences and Engineering, King Abdullah University of Science and Technology, Saudi Arabia. His research interests include online social network measurement and modeling.

Bruno Ribeiro received the Ph.D. degree in computer science from University of Massachusetts Amherst. He is currently an assistant professor in the Department of Computer Science at Purdue University. His central research interest is the principled measurement, analysis, and mining of large-scale complex social and communication networks.



John C. S. Lui received the Ph.D. degree in computer science from UCLA. He is currently a professor in the Department of Computer Science and Engineering at The Chinese University of Hong Kong. His current research interests include communication networks, networksystem security, network economics, network sciences, cloud computing, large-scale distributed systems and performance evaluation theory.



Don Towsley holds a B.A. in Physics (1971) and a Ph.D. in Computer Science (1975) from University of Texas. From 1976 to 1985 he was a member of the faculty of the Department of Electrical and Computer Engineering at the University of Massachusetts, Amherst. He is currently a Distinguished Professor at the University of Massachusetts in the Department of Computer Science. His research interests include network measurement and modeling.

Xiaohong Guan received the B.S. and M.S. degrees in automatic control from Tsinghua University, Beijing, China, in 1982 and 1985, respectively, and the Ph.D. degree in electrical engineering from the University of Connecticut, Storrs, US, in 1993. He is currently a professor at the Systems Engineering Institute, Xi'an Jiaotong University, Xi'an, China. His research interests include allocation and scheduling of complex networked resources, network security, and sensor networks.