2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining

Unbiased Sampling of Social Media Networks for Well-connected Subgraphs

Dong Wang⋆, Zhenyu Li⋆, Gareth Tyson‡, Zhenhua Li†, Gaogang Xie⋆
⋆ICT-CAS, ‡ Queen Mary University of London, †Tsinghua Univ.
{wangdong01, zyli, xie}@ict.ac.cn, g.tyson@qmul.ac.uk, lizhenhua1983@gmail.com

***Abstract*—Sampling social graphs is critical for studying things like information diffusion. However, it is often necessary to laboriously obtain *unbiased* and *well-connected* datasets because existing survey algorithms are unable to generate well-connected samples, and current random-walk based unbiased sampling algorithms adopt rejection sampling, which heavily undermines performance.This paper proposes a novel random-walk based algorithm which implements Unbiased Sampling using Dummy Edges (USDE). It injects dummy edges between nodes, on which the walkers would otherwise experience excessive rejections before moving out from such nodes. We propose a rejection probability estimation algorithm to facilitate the construction of dummy edges and the computation of moving probabilities. Finally, we apply USDE in two real-life social media: Twitter and Sina Weibo. The results demonstrate that USDE generates well-connected samples, and outperforms existing approaches in terms of sampling efficiency and quality of samples.**

I. Introduction

Social media have gained tremendous popularity in recent years. In order to characterize, optimize and simulate information diffusion within such networks, it is often necessary to collect realistic datasets [1][2]. However, the huge size of these networks makes it very hard to gain a true snapshot of the complete graph. Hence, it becomes necessary to obtain an *unbiased* and *well-connected* subgraph of the network. Here, a subgraph sample is unbiased if every user in the social media is sampled with equal probability, as widely adopted in literature [3][4]; and a sample is said to be well-connected if it has only a few connected components.

Sampling networks has been heavily studied in the literature (§II). Survey sampling approaches (like stratified sampling [5] or uniform sampling [6]) and those using random jumps during sampling [4], while being able to provide unbiased estimations of individual node attributes (like node degree), fail to generate well-connected samples. Another kind of sampling, called random-walk sampling, while being able to provide well-connected samples, generates biased towards high-degree nodes. Several algorithms, including Metropolis-Hastings Random Walk (MHRW) [3] and its variants [7][8], were proposed to adapt random-walk based sampling to generate unbiased samples using the *rejection sampling* procedure.

The rejection sampling procedure explicitly rejects moving to high-degree nodes by increasing the probability of (re)sampling low-degree nodes. This procedure incurs the cost and delay of sampling without gathering information in exchange, especially when applied to online social media with local disassortative mixing pattern, where users tend to follow power-users (like celebrities) with degrees that are orders of magnitude larger [9][10]. In such networks, the random walkers will be trapped in the low-degree nodes for a long time due to the rejection sampling of high-degree nodes, and meanwhile the low-degree nodes will be repeatedly sampled wastefully. Unfortunately, if these repetitions are removed, MHRW and its variants using rejection sampling fail to provide unbiased samples [11].

The above facts motivate us to propose a new sampling algorithm (§III): Unbiased Sampling with Dummy Edges (*USDE*). We aim to provide a good quality of samples (measured by the ability to provide unbiased and well-connected subgraphs) with high sampling efficiency (measured by the sampling convergence and the speed of discovering new nodes). USDE is a random-walk based algorithm that exploits artificially injected *dummy edges* (§III-A). The intuition is that, instead of rejecting high-degree nodes, the walkers move through dummy edges to other nodes, on which the walkers would otherwise experience excessive rejections. Note that the dummy edges are not included in the final samples and are only used in the sampling process temporarily. We propose a rejection probability (also called *self-sampling probability*) estimation algorithm to facilitate the construction of dummy edges and the computation of traversal (moving) probabilities (§III-B). By carefully assigning moving probabilities between adjacent nodes that are connected by either original edges or dummy edges, we show that our algorithm is able to generate unbiased and well-connected samples. The evaluation on two real-life social media, Twitter and Sina Weibo, further validates the efficiency (§IV).

II. Background and Motivation

A. Background

Due to a low level of reciprocity [10], social media are usually abstracted as directed graphs. However, from the perspective of sampling, it has been shown in [11][8] that a social media network can be viewed as an undirected graph:

ASONAM '17, July 31 - August 03, 2017, Sydney, Australia

ACM ISBN 978-1-4503-4993-2/17/07?/$15.00
http://dx.doi.org/10.1145/3110025.3110141

$G = (V, E)$, where V is the set of nodes representing users, and E is the set of bidirectional edges that are obtained by treating unidirectional edges as bidirectional ones. Another prominent feature of online social media is that nodes with a very high degree (*e.g.* celebrities) may be surrounded by many low-degree nodes [10][9], implying *high local disassortativity*. Here, dissasortativity captures the phenomenon that nodes tend to be connected with other nodes with different degrees [12].

B. Motivation

The existence of local disassortative nodes makes unbiased sampling challenging. Considering a high-degree node u that it is surrounded by many low-degree nodes, crawlers in the native random-walk sampling will move from u to its low-degree neighbor v with a low probability. And once the crawlers reach low-degree nodes, the native sampling prefers high-degree nodes as the next hop. This leads to sampling bias towards high-degree nodes. To address this problem, two kinds of methods can be adopted. One is using random jumps [11], [4], *i.e.* allowing the crawler to jump to a randomly chosen node (instead of following edges). Random jumps essentially decrease the sampling probability of high-degree nodes, while increase the probability of low-degree ones. Random jumps, however, will yield many small and isolated components in the final samples.

The other solution is using rejection sampling, like MHRW [3], *i.e.* rejecting to move to high-degree nodes but sampling low-degree nodes many times (called *self-sampling* here). Self-sampling increases the sampling probability of low-degree nodes and implicitly reduces the probability of high-degree nodes. However, for a network with local disassortativity, the self-sampling probability of low-degree nodes will be extremely high, because crawlers must experience excessive rejections of moving towards high-degree neighbors in order to compensate the sampling probabilities of low-degree nodes. The excessive rejections greatly hurts the efficiency of discovering new nodes.

III. Unbiased Sampling with Dummy Edges

This section presents our proposed sampling algorithm, USDE. The idea is to keep the connectivity and unbiased nature of samples obtained by rejection sampling algorithms (like MHRW), while avoiding excessive rejections (or self-samplings). To this end, we add dummy edges between nodes that would otherwise experience high self-samplings, and amortize the self-sampling probabilities of individual nodes to moving probabilities on the dummy edges. Importantly, dummy edges are only used during the sampling process to allow the sampling crawler to move to another node, and they are not involved in the final sample. Table I lists the notations used throughout of this paper.

A. Dummy Edges

Dummy edges are built between nodes that experience extensive self-sampling (rejection) probabilities in random-walk based rejection samplings. Suppose the crawler is currently on node i; the candidate nodes for building dummy edges from i are the previously visited nodes' neighbors that have not been visited yet *and* have non-zero self-sampling probabilities. In this way, a node that connects with i through a dummy edge once being visited cannot form new connected components, since at least one of its neighbors has been visited. This avoids jumping to random nodes, and therefore avoids the generation of many small components.

TABLE I: Notations in the description of USDE

Notation	Definition
V'	set of nodes having been visited by sampling walkers
k_i	degree of node i in the abstracted undirected network
$P_{i,j}$	moving probability from node i to j through one step
$P_{i,j}^{(n)}$	moving probability from node i to j through n steps
π_i	probability that node i can be sampled
L_i	self-sampling probability of node i, *i.e.* $P_{i,i}$
$U(i)$	set of nodes that connect with i through dummy edges
$S(i)$	set of nodes that connect with i through original edges
$DP_{i,j}$	moving probability from i to j via the dummy edge
LP_i	lower bound of node i's self-sampling probability
$E_r(k)$	the average node degree with sampling repetitions
$E_u(k)$	the average node degree without sampling repetitions

B. Moving Probability in USDE

At first, we describe the calculation of the moving probability from node i to node v in USDE using Eq. 1:

$$P_{i,v} = \begin{cases} min(\frac{1}{k_v}, \frac{1}{k_i}) & \text{if } v \in S(i) \\ DP_{i,v}, & \text{if } v \in U(i) \\ 1 - \sum_{x \in S(i)} P_{i,x} - \sum_{y \in U(i)} P_{i,y} & \text{if } v = i \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $U(i)$ is the set of nodes that have dummy edges with i, $S(i)$ is the set of original neighbors of i, and $DP_{i,j} = DP_{j,i}$.

We have showed in our previous work [7] that the sufficient and necessary condition for unbiased (nodal) sampling in a network G that has at least one node with non-zero clustering coefficient is: **(1)** if $P_{i,j} > 0$, then $P_{j,i} > 0$; **(2)** $\forall j \in V, \sum_{i=1}^{|V|} P_{i,j} = 1$, where $|V|$ is the number of nodes in network G. The above moving probability of USDE meets these two conditions, because following Eq. 1: $P_{i,j} = P_{j,i}$ and thus if $P_{i,j} > 0$, then $P_{j,i} > 0$; and $\forall i \in V, \sum_{j=1}^{|V|} P_{j,i} = \sum_{j=1}^{|V|} P_{i,j} = 1$. Hence, USDE generates unbiased (nodal) samples where each node is sampled with equal probability.

C. Dummy Edge Addition

The selection of nodes to build dummy edges is critical in USDE. The nodes to which dummy edges can be built from the currently visited node are the previously visited nodes' neighbors that have a non-zero self-sampling probability *and* have not yet been visited by the sampling crawler. However, finding such nodes is challenging because we are unable get the exact self-sampling probability of an unvisited node. Instead, we estimate the lower bound of the self-sampling

probability for an unvisited neighbor based on the degree of the current visited node and the degree of this neighbor.

From Eq. 1, we can infer that if there is a neighbor u of node i with degree $k_u > k_i$, then the self-sampling probability of i without dummy edges is at least $(\frac{1}{k_i} - \frac{1}{k_u})$. Following this observation, the lower bound of an unvisited node i's self-sampling probability without dummy edges, LP_i, is:

$$LP_i = \sum_{v \in \{S(i) \cap V'\}} (\frac{1}{k_i} - \frac{1}{k_v}) \quad \text{where } k_v > k_i \tag{2}$$

where $S(i)$ is the neighbor set of i and V' is the set of nodes that have been visited.

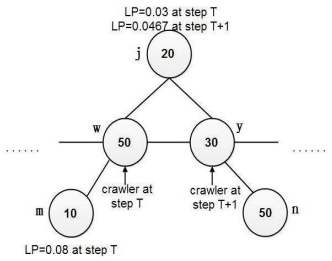


Fig. 1: Estimating self-sampling probability

Fig. 1 illustrates how the lower bounds are estimated during the sampling process. The node degrees are marked on the nodes and the node IDs are labeled beside the nodes. Let's assume that, before step T, the self-sampling probability for all nodes is 0. At step T, the crawler visits node w. Node j and node m are neighbors of w and $k_j < k_w$, $k_m < k_w$. Hence, we can estimate that $LP_j = \frac{1}{k_j} - \frac{1}{k_w} = \frac{1}{20} - \frac{1}{50} = 0.03$ and $LP_m = \frac{1}{k_m} - \frac{1}{k_w} = \frac{1}{10} - \frac{1}{50} = 0.08$ at step T. At the next step $T+1$, the crawler visits y. We update the estimation of LP_j because j is also a neighbor of y and $k_j < k_y$. The LP value of j is updated as $LP_j = 0.03 + \frac{1}{20} - \frac{1}{30} = 0.0467$. We are unable to estimate LP_n at $T+1$ as $k_n > k_y$.

During the process of sampling, we use a queue Q to record the node ID and the estimated lower bound of self-sampling probability, LP_v, for each unvisited node v with $LP_v > 0$. The nodes in this queue are candidate nodes for dummy edge addition from the currently visited node.

D. Computation of moving probabilities

When a node i is visited for the first time, we obtain its self-sampling probability as $1 - \sum_{x \in S(i)} P_{i,x} - \sum_{y \in U(i)} P_{i,y}$ according to Eq. 1. $U(i)$ is empty in the case that i has never been selected for building dummy edges before. If such a probability is larger than a threshold δ, we pop a tuple (v, LP_v) from Q and add a new dummy edge between node i and v. The moving probability of the added dummy edge $DP_{i,v}$ ($DP_{v,i}$) is computed as Eq. 3.

$$DP_{i,v} = min(LP_v, 1 - \sum_{x \in S(i)} P_{i,x} - \sum_{y \in U(i)} P_{i,y}) \tag{3}$$

We then update LP_v with $LP_v^{'} = LP_v - DP_{v,i}$. If the updated $LP_v^{'} > 0$, we push the updated tuple $(v, LP_v^{'})$ back to Q. Such an estimation method (for the lower bounds of self-sampling probability) ensures that the sampled subgraphs are well-connected because all the nodes recorded in Q are neighbors of previously sampled ones.

The moving probability $DP_{i,v}$ ($DP_{v,i}$) of the dummy edge can be significant in the case of large LP_v and self-sampling probability. In this case, the sampling crawler will wander between i and v for a long time, which prevents the crawler from finding new nodes. To solve this problem, rather than pop only 1 tuple, we pop γ ($\gamma > 1$) tuples $(v_1, LP_{v_1}), (v_2, LP_{v_2}), (v_3, LP_{v_3}) \cdots (v_\gamma, LP_{v_\gamma})$ from Q at one time, and γ dummy edges are added $\{i, v_j\}$ ($j = 1, 2 \cdots \gamma$). The moving probability on dummy edge $\{i, v_j\}$ DP_{i,v_j} ($DP_{v_j,i}$) is computed as Eq. 4.

$$DP_{i,v_j} = min(LP_{v_j}, \frac{1 - \sum_{x \in S(i)} P_{i,x} - \sum_{y \in U(i)} P_{i,y}}{\gamma}) \tag{4}$$

LP_{v_j} is then updated accordingly and the tuples with non-zero LPs are pushed back to Q. The addition of dummy edges ceases when the self-sampling probability on i is reduced to 0, or dummy edges to all the γ nodes are added.

The queue Q applies FIFO (First In First Out), which avoids adding too many dummy edges on a single node. The stored nodes with fewer dummy edges and larger LP are more likely to be popped out, reducing self-sampling probabilities as much as possible. The queue Q is initialized in the first t iterations of sampling. During this time period, dummy edges are not added, and the LPs of the visited nodes' neighbors are estimated and pushed into Q as dummy edge candidates.

IV. Sampling Twitter and Sina Weibo

To confirm the practicality of our technique, we use USDE to sample Twitter and Sina Weibo, and focus on the quality of samples and the sampling efficiency.

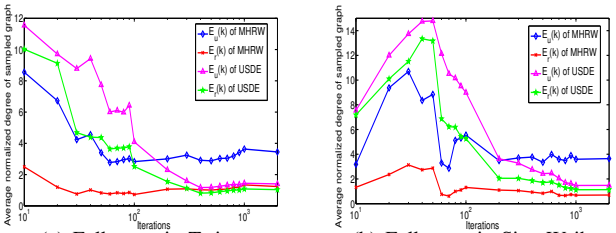
A. Experiment Setup

Both Twitter and Sina Weibo have a dense numerical ID space. We randomly generate a number of IDs, and choose 10 IDs that correspond to valid users as the initial seeds for crawlers. Since the relationship between users might be non-reciprocal in Twitter and Sina Weibo, we leverage the idea of "backward edge traversals" [8], where unidirectional edges are treated as bidirectional ones.

The evaluation of unbias requires the ground truth of user attributes in Twitter and Sina Weibo. Since we are aiming at unbias in terms of nodes (as opposed to edges), the ground truth of user attributes can be estimated using UNI [3]. We thus uniformly generate a large number of user IDs at random and use these IDs to query Twitter and Sina Weibo APIs. During the sampling process, we rely on the APIs provided by Twitter and Sina Weibo to obtain the information of the currently visited node's neighbors.

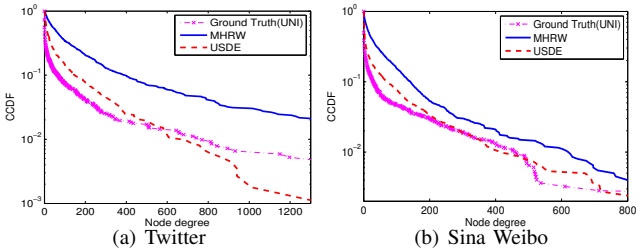
B. Quality of samples

We first examine the accuracy of estimating the number of followers in Fig. 2, where the y-axis shows the average number of followers (normalized by the ground truth). Due to



(a) Followers in Twitter (b) Followers in Sina Weibo

Fig. 2: Normalized average number of followers



(a) Twitter (b) Sina Weibo

Fig. 3: Complementary CDF of followers in samples

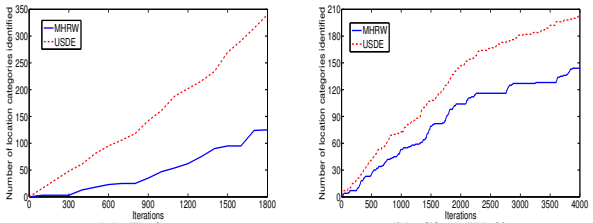
the existence of self-samplings, we measure two node degree metrics: $E_r(k)$ — the average node degree when m samplings on a node are counted as m nodes with identical properties; and $E_u(k)$ — the average node degree when a node is only counted once no matter how many times it is sampled. We observe large variations of $E_r(k)$ and $E_u(k)$ for the initial sampling iterations due to the random choice of seed nodes. The normalized $E_r(k)$ gradually converges to 1 after taking several hundred iterations. Nevertheless, while the normalized $E_u(k)$ estimated by USDE in both networks converges close to 1, the $E_u(k)$ estimated by MHRW is about four times as high as the ground truth. Such a huge difference is due to excessive sampling repetitions in MHRW.

Fig. 3 further examines the distribution of followers obtained by sampling algorithms after 2,000 iterations, where the sampled users are counted uniquely. It is notable that USDE generates a much closer distribution to UNI (ground truth for nodal properties) than MHRW. For instance, samples generated by USDE reveal 93% of users have fewer than 200 followers, close to the the corresponding proportion of ground truth 95%. Nevertheless, this percentage in samples generated by MHRW is only 80%.

C. Sampling Efficiency

Our results reveal that, for both Twitter and Sina Weibo, the average sampling times per node by USDE is close to 2, much lower than that by MHRW, which is 6-8 and 8-10 for Twitter and Sina Weibo, respectively.

We finally investigate the efficiency in identifying new user locations in Fig. 4. USDE consistently identifies many more new locations than MHRW. For example, USDE identifies more than 200 cities in Twitter within 1,200 iterations, while MHRW only identified about 50. The difference between the two curves in Twitter is more significant than that in Sina



(a) Twitter (b) Sina Weibo

Fig. 4: Efficiency of identifying location information

Weibo, possibly due to the categories of attributes in Twitter (world-wide), which are more diverse than the ones in Sina Weibo (country-wide).

V. Conclusion

In this paper, we propose a novel random-walk based sampling algorithm: USDE. It introduces dummy edges between nodes with high self-sampling probabilities to allow crawlers to move between nodes, while still keeping the connectivity of samples. We have detailed the way of building dummy edges and the computation of moving probabilities. The evaluations on two real-life social media have demonstrated that, in comparison with existing algorithms, USDE achieves a better quality of samples and higher sampling efficiency.

Acknowledgement

This work was supported in part by National High-tech R&D Program of China No. 2015AA010201 and 2015AA01A201, by the National Natural Science Foundation of China No. 61572475 and 61502460.

References

[1] D. Wang, H. Park, G. Xie, S. Moon, M.-A. Kaafar, and K. Salamatian, "A genealogy of information spreading on microblogs: A galton-watson-based explicative model," in *Proceedings IEEE INFOCOM*, 2013.

[2] D. Wang, Z. Li, G. Xie, M. A. Kaafar, and K. Salamatian, "Adwords management for third-parties in sem: An optimisation model and the potential of twitter," in *Proceedings of IEEE INFOCOM*, 2016.

[3] M. Gjoka, M. Kurant, C. T. Butts, and A. Markopoulou, "Walking in facebook: A case study of unbiased sampling of osns," in *Proceedings of the IEEE INFOCOM*, 2010.

[4] B. Ribeiro, P. Wang, F. Murai, and D. Towsley, "Sampling directed graphs with random walks," in *Proceedings of IEEE INFOCOM*, 2012.

[5] C.-E. Sarndal, "Stratified sampling," in *Model Assisted Survey Sampling*. Springer, 2003.

[6] M. R. Henzinger, A. Heydon, M. Mitzenmacher, and M. Najork, "On near-uniform url sampling," *Computer Networks*, vol. 33, no. 1, 2000.

[7] D. Wang, Z. Li, and G. Xie, "Towards unbiased sampling of online social networks," in *Proceedings of the IEEE ICC*, 2011.

[8] T. Wang, Y. Chen, Z. Zhang, P. Sun, B. Deng, and X. Li, "Unbiased sampling in directed social graph," *ACM SIGCOMM Computer Communication Review*, vol. 41, no. 4, 2011.

[9] M. Cha, H. Haddadi, F. Benevenuto, and P. K. Gummadi, "Measuring user influence in twitter: The million follower fallacy." in *Proceedings of ICWSM*, 2010.

[10] H. Kwak, C. Lee, H. Park, and S. Moon, "What is twitter, a social network or a news media?" in *Proceedings of WWW*, 2010.

[11] B. F. Ribeiro and D. Towsley, "On the estimation accuracy of degree distributions from graph sampling," in *Proceedings of IEEE Decision and Control (CDC)*, 2012.

[12] M. Piraveenan, M. Prokopenko, and A. Zomaya, "Local assortativeness in scale-free networks," *EPL (Europhysics Letters)*, vol. 89, no. 4, p. 49901, 2010.