Estimating Sizes of Social Networks via Biased Sampling

Liran Katzir, Edo Liberty, and Oren Somekh
Yahoo! Labs., Haifa, Israel
{lirank, edo, orens}@yahoo-inc.com

ABSTRACT

Online social networks have become very popular in recent years and their number of users is already measured in many hundreds of millions. For various commercial and sociological purposes, an independent estimate of their sizes is important. In this work, algorithms for estimating the number of users in such networks are considered. The proposed schemes are also applicable for estimating the sizes of networks' sub-populations.

The suggested algorithms interact with the social networks via their public APIs only, and rely on no other external information. Due to obvious traffic and privacy concerns, the number of such interactions is severely limited. We therefore focus on minimizing the number of API interactions needed for producing good size estimates. We adopt the abstraction of social networks as undirected graphs and use random node sampling. By counting the number of collisions or non-unique nodes in the sample, we produce a size estimate. Then, we show analytically that the estimate error vanishes with high probability for smaller number of samples than those required by prior-art algorithms. Moreover, although our algorithms are provably correct for any graph, they excel when applied to social network-like graphs. The proposed algorithms were evaluated on synthetic as well real social networks such as Facebook, IMDB, and DBLP. Our experiments corroborated the theoretical results, and demonstrated the effectiveness of the algorithms.

Categories and Subject Descriptors

F.2.2 [**Theory of Computing**]: Analysis of Algorithms and Problem Complexity—*Nonnumerical Algorithms and Problems*

General Terms

Algorithms

Keywords

Population Estimator, Sampling, Undirected Graph, Social Network

WWW2011 Mar. 28 - Apr. 1, Hyderabad, India

1. INTRODUCTION

Online social networks have become increasingly popular in the recent decade which gave rise to an increasing need in analyzing their properties and comparing them to one another. Many properties of online social networks are considered important. These include, for example: their user age distribution, net activity, connectivity, and many more. The literature on attaining such parameters is vast and any effort to reference all of it is bound to fail. We cite here only a handful of directly related works as examples. In [12, 9] the authors present a way to detect proximity between two users. In [19] the authors analyze the degree distribution, clustering property, degree correlation, and evolution over time for three networks. That said, the total number of users (or users in a certain demographic) seems to be one of the most crucial factors in deriving the worth and overall performance of social networks. These figures are also critical for business development issues, choosing between networks for advertisement campaigns or for launching social applications. Although countless sites, blogs, and other reports often present such numbers, these are usually based on reports of the social networks themselves or on traffic analysis and are not guaranteed to be accurate (e.g. [2]).[1] Moreover, since each network reports slightly different figures, it is almost impossible to compare between them. For example, Facebook reported lately on crossing the 500 million active users mark. However, it is unclear what constitutes as an "active user". Thus, it is extremely important to be able to accurately and reliably estimate the size of social networks in a unified and unbiased way (without the networks' consent or control).

Since online social networks provide public interfaces, it is possible to traverse their members' network externally. By "crawling" the network, we can collect statistics on any of the above characterizes. In a similar spirit, in [3, 4] the authors suggested ways to measure various parameters of search engines by interacting with their public search interface only. One approach, is to crawl the network extensively. In other words, since online social networks can be modeled as undirected unweighted graphs (where the users are nodes and edges are connections/friendships) we can perform a *breadth-first search* (BFS) on the graph.[2] This method is

[1]Moreover, the published statistics do not provide an estimate for connected sub-graphs (For example, 20-30 year olds living at the US).

[2]Note that online social networks' public APIs provide lists of connected users for every user, Thus, acting like a neighbor list representation of the graph.

impractical when dealing with online social networks since the communication and computational burden of such an undertaking would probably be prohibitive.

The second approach is to sample users uniformly at random from the network. From a uniform sample, most statistics can be estimated. This includes estimating the size of the network using methods such as *mark-and-recapture*. These methods have the advantage of requiring only $O(\sqrt{n})$ users to be sampled to get a good estimate (where n is the overall number of users). Their disadvantage though, is that users must be sampled uniformly. Since online social networks interfaces usually do not provide this functionality, it must be simulated by other API queries which give for each user the list of their neighbors. Alas, producing a single uniformly chosen user might require many such queries. This is explained in the next section.

In this work we show that there is no need to sample users uniformly. In fact, by using the sampling bias we reduce the number of required samples dramatically. For example, for networks with a Zipfian-like degree distribution our algorithms require only $O(n^{1/4}\log(n))$ samples to converge. Moreover, since the bias does not have to be corrected, each such sample requires considerably less API queries. Surprisingly, our algorithm is extremely simple and gives provable error guarantees with high probability.

Experiments with our algorithm performed over a wide range of real and synthetic data corroborate that it converges significantly faster and to a more accurate estimate than uniform sampling based approaches.

As a side note, simple variants of our algorithm also give efficient sub-linear algorithm for estimating the size of transitive closures in graphs and for estimating the size of search-indexes as in [3]. However, this is a matter of farther research and is beyond the scope of this paper.

The rest of this work is organized as follows. Section 2 surveys related work. Our algorithms are presented and analyzed in Sections 3 and 4. In Section 5 we report our experimental results and conclude in Section 6. Various proofs and discussions are included in the Appendix.

2. BACKGROUND AND PRIOR WORK

From this point on we consider the general problem of estimating the size of undirected graphs. The graph representation of online social networks is the obvious one. Each node refers to one user and an edge is present between two nodes if their corresponding users are "friends" in the social network. Although our algorithms are correct for general graphs, they are especially suited for graphs which naturally occur in large social networks.

In [16] the authors provide a possible solution for another problem which could be used to solve the problem at hand as well. They present an algorithm for estimating the number of attributes in a database. The algorithm samples rows from the database uniformly at random. It then estimates the total number of attributes using the collected information of how many times each attribute was picked. This is identical to a well known problem in statistics called "estimating the number of unseen types". Their algorithm can be applied to estimating the size of graphs if the graph is represented as a database table containing two rows per edge, one for each adjacent node.[3] Clearly, the number of distinct attributes in this database is n, the number of nodes. The algorithm presented in [16] is guaranteed to take at most $r = O(n)$ samples. However, in graphs, unlike in databases, it is possible to sample nodes (analogously, attributes) uniformly at random which can dramatically decrease the number of samples.

In ecology, a method known as *mark and recapture* is used to estimate population sizes.[4] It relies on the same phenomenon as the so called "birthday paradox" effect. Informally, after sampling r nodes uniformly at random we expect to encounter $C \approx r^2/2n$ collisions (nodes already picked). Thus, n can be estimated by $r^2/2C$. Surprisingly, taking only $O(\sqrt{n})$ samples can guarantee that this estimate for n is rather accurate.[5] In [7] the authors present a maximum likelihood estimator for this problem and show that their estimator converges almost surly when the number of samples increases. In [6, 10] the authors extend these methods to non-uniform, but known, distributions.

That said, to use these methods one must sample nodes uniformly at random from a graph, which is not straight forward. To see how this can be done we remind the reader a few basic facts from spectral graph theory. A random walk on an undirected graph with n nodes $\{v_1, \ldots, v_n\}$ is defined as such: start from an arbitrary node, then move to a neighboring node uniformly at random and repeat. After many such steps, the probability of being at any node v_i is close to $p_i = d_i/D$ where d_i is the degree of node v_i and $D = \sum_{i=1}^{n} d_i$ is the sum of all node degrees in the graph. This is called the stationary distribution of the random walk on the graph. The number of random walk steps needed for the stationary distribution to be reached depends on the mixing rate property of the graph (see a survey by Lovász [13]). Fortunately, social network graphs and small world graphs are known to have good mixing rates. We therefore assume that nodes can be repeatedly sampled from the stationary distribution without much computational overhead.

Using these properties of random walks, one can sample nodes also uniformly at random by using, for example, rejection sampling. To be precise, a node v_i is first sampled according to the stationary distribution. Then, with probability $1/d_i$ it is kept. With probability $1 - 1/d_i$ it is rejected. Clearly the set of kept nodes is uniformly sampled. However, since we only expect to accept a node with probability n/D, to sample $\Omega(\sqrt{n})$ un-rejected nodes would require an expected $r = \Omega(D/\sqrt{n})$ biased samples. Several rejection sampling ideas and other methods for turning the node sampling distribution to uniform were suggested for specific graphs. Namely, the bipartite graph between search queries and search results [5, 3].[6] Another approach was considered in [8]. The authors present a modified Metropolis-Hastings random walk which transitions from node v_i to an adjacent node v_j with probability $1/\max(d_i, d_j)$. With the remaining probability, it stays in v_i. Due the symmetry in the transition probabilities it can be shown that the stationary distribution of this walk is uniform on the nodes. However,

[3]Sampling uniformly from this table is possible in our setting since random walks on graphs sample edges uniformly.

[4]Other names for this method or closely related ones, include capture-recapture, capture-mark-recapture, mark-recapture, and mark-release-recapture.

[5]We make a more general statement later in this paper.

[6]In fact, the authors try to compare between two different search services but their approach is suitable for this task as well.

the mixing rate of this walk can be significantly worse than that of the original graph, and so, it is unclear when it is expected to outperforms rejection sampling, i.e., require fewer random walk steps.

An interesting tangentially related problem is known as the "German tank problem" [1]. It was supposedly used during world-war II to estimate the number of German tanks based on manufacturing numbers found on those captured by the allied forces. In its mathematical formulation, elements with serial ID's are sampled uniformly without replacement and the objective is to provide an estimate for the total number of elements. This is not applicable to our scenario since the users do not have serially allocated and publicly available ID's.

Estimating the number of nodes in a graph was also studied. In [11] the authors estimate the size of a tree. Their motivation was to estimate the running time of a backtracking programs. Later [17] extends their argument to acyclic graphs. Finally [14] extends this idea to general undirected graphs. However, the running time of the the latter is unbounded in the worst case and expected to be more than the number of nodes in the graph. Recently, in [18] the authors try to estimate the size of social networks in a setup very similar to ours'. However, they either require that the users be sampled uniformly or use the algorithm from [14] which their experiments show is impractical.

3. COLLISION COUNTING

In this section we present our graph size estimator. We start by taking r samples $\{x_1, \ldots, x_r\}$ independently from the stationary distribution of the graph, i.e., each node v_i is sampled with probability $p_i = d_i/D$ where $D = \sum_{i=1}^{n} d_i$.

We define three variables that the algorithm keeps track of. First, the number of collisions. A collision is a pair of identical samples. More precisely, define $Y_{i,j}$ to be 1 if $x_i = x_j$ and 0 else. $C = \sum_{i<j} Y_{i,j}$. Second, Ψ_1 is the sum of all sampled node degrees $\Psi_1 = \sum_{i=1}^{r} d_{x_i}$. Last, we define Ψ_{-1} as the sum of the reciprocal degrees, $\Psi_{-1} = \sum_{i=1}^{r} 1/d_{x_i}$.

Using linearity of expectation and the fact that the samples are taken independently it is easy to compute their expectation.

$$\begin{aligned}
\mathbb{E}[C] &= \mathbb{E}[\sum_{i<j} Y_{i,j}] = \binom{r}{2}\mathbb{E}[Y_{i,j}] = \binom{r}{2}\sum_{i=1}^{n} p_i^2 \\
\mathbb{E}[\Psi_1] &= \mathbb{E}[\sum_{i=1}^{r} d_{x_i}] = r\mathbb{E}[d_{x_1}] = r\sum_{i=1}^{n} p_i d_i = rD\sum_{i=1}^{n} p_i^2 \\
\mathbb{E}[\Psi_{-1}] &= \mathbb{E}[\sum_{i=1}^{r} 1/d_{x_i}] = r\mathbb{E}[1/d_{x_1}] = r\sum_{i=1}^{n} p_i/d_i = \frac{rn}{D} .
\end{aligned}$$

From the above three equations we can isolate n and get that:

$$n = \binom{r}{2}\frac{\mathbb{E}[\Psi_1]\mathbb{E}[\Psi_{-1}]}{r^2\mathbb{E}[C]} \approx \frac{\mathbb{E}[\Psi_1]\mathbb{E}[\Psi_{-1}]}{2\mathbb{E}[C]}$$

Intuitively, if C, Ψ_1 and Ψ_{-1} are all close to their expected values then the estimator $\Psi_1\Psi_{-1}/2C$ should be close to n as well.[7] This is stated in Corollary 1.

[7]It is also possible to use $\mathbb{E}[\Psi_1\Psi_{-1}]$ directly and get a slightly different estimator. Namely, $\hat{n} = (\Psi_1\Psi_{-1} - r)/2C$. This however is different from $\Psi_1\Psi_{-1}/2C$ only by a factor of $O(r/nc)$.

COROLLARY 1. *For any degree distribution and C, Ψ_1, and Ψ_{-1} defined as above we have the estimator:*

$$\hat{n} \triangleq \frac{\Psi_1\Psi_{-1}}{2C} \tag{1}$$

which guarantees for any $\epsilon \le 1/2$ and $\delta \le 1$:

$$\Pr[n(1-\epsilon) \le \hat{n} \le n(1+\epsilon)] \ge 1-\delta$$

as long as the number of samples, r, satisfies:

$$r \ge r_c \in O\left\{\frac{1}{\epsilon\sqrt{\delta}\sqrt{\sum_{i=1}^{n} p_i^2}}, \frac{\sum_{i=1}^{n} p_i^3}{\epsilon^2\delta(\sum_{i=1}^{n} p_i^2)^2}, \frac{\sum_{i=1}^{n} 1/p_i}{\epsilon^2\delta n^2}\right\} .$$

PROOF. The proof goes by first calculating the variance of C, Ψ_1 and Ψ_{-1}. Then, using Chebyshev's inequality we require that each of them gives an $\epsilon/4$ approximations to their expected values with probability at least $1-\delta/3$. This gives us a lower bound on the required number of samples. See Appendix A for more details. Since the probability of each for deviating from its expected value is at most $\delta/3$, the probability of one or more of them deviating is at most δ (the union bound). This gives us that with probability at least $1-\delta$ all three variables are $\epsilon/4$ close to their respective expectations. We then use the facts that $\hat{n}/n \le (1+\epsilon/4)^2/(1-\epsilon/4) \le 1+\epsilon$ and $\hat{n}/n \ge (1-\epsilon/4)^2/(1+\epsilon/4) \ge 1-\epsilon$ to complete the proof for $\epsilon \le 1/2$. □

Note that for regular graphs, where the degree distribution is uniform, this estimator is identical to the "birthday paradox" estimator presented in the introduction. Moreover, it will also require $O(\sqrt{n})$ samples to converge (to see this substitute $1/n$ for p_i).

3.1 Performance for online social networks

In order to argue that our algorithm is suitable for sizing social networks we have to assume something about their node degree distributions. In [15], [19] and in [8] the authors argue that, in several networks, the nodes' degrees exhibits different kinds of heavy tail distributions. Mainly: Exponential, Double-Pareto and Zipfian. Here we analyze, as an example, the Zipfian distribution. Similar analyses can be performed for the other distributions as well.

If the nodes' degrees are distributed according to a Zipfian distribution with maximum degree of d_m and parameter $\alpha = 2$ we have:

$$Pr(d = j) = \frac{j^{-2}}{H} \quad ; \quad j = 1, \ldots, d_m ,$$

where $H = \sum_{j=1}^{d_m} j^{-2} \approx \frac{\pi^2}{6}$ and $1 \ll d_m = \Theta(\sqrt{n})$. The ℓ'th moment of the degree distribution is defined as $\mathcal{M}_\ell = \mathbb{E}[d^\ell]$ and the first few moments of the Zipfian distribution are:

$$\mathcal{M}_{-1} \le \frac{1}{H}; \quad \mathcal{M}_1 \approx \frac{\log d_m}{H}; \quad \mathcal{M}_2 \approx \frac{d_m}{H}; \quad \mathcal{M}_3 \approx \frac{d_m^2}{2H} .$$

We also assume that the moments of the observed degree distribution are close to those of the generating distribution. This is true for large graphs by the strong law of large numbers (SLLN). This gives us that $\sum_{i=1}^{n} p_i^\ell \approx \frac{\mathcal{M}_\ell}{n^{\ell-1}(\mathcal{M}_1)^\ell}$. Substituting the above into Corollary 1 and using the fact that $d_m = \Theta(\sqrt{n})$ we get that $r_c \in O(n^{1/4}\log(n))$. Therefore, only $O(n^{1/4}\log(n))$ samples suffice for our estimator to be accurate. Note the significant reduction in the number of samples over the uniform distribution. For example, for $n = 10^9$, $\sqrt{n} \approx 30,000$ while $n^{1/4}\log(n) \approx 6000$.

3.2 Subgraph size estimation

One surprising aspect of this estimator is that it works for subgraphs as well. Let X' be the subset of samples X who are also in the subgraph. We perform the same random walk and compute the same parameters C', Ψ'_1 and Ψ'_{-1}, which are defined as above but for X' instead of X. The subgraph size is estimated by $\Psi'_1\Psi'_{-1}/2C'$. The proof provided above works for this case as well. The only change is that D is replaced by D' which is the sum of node degrees in the subgraph. However, since D (and therefore D' as well) cancels itself in the analysis our estimator remains unchanged.

However, it turns out that it is usually more efficient to first estimate the size of the entire graph and then estimate the subgraphs' relative size. From the above we have that $\mathbb{E}[\Psi_{-1}] = rn/D$, similarly for the subgraph, $\mathbb{E}[\Psi'_{-1}] = r'n'/D'$ (n' being the number of nodes in the subgraph). Isolating n' we get:

$$n' = n\frac{r}{r'}\frac{D'}{D}\frac{\mathbb{E}[\Psi'_{-1}]}{\mathbb{E}[\Psi_{-1}]} \approx n\frac{\Psi'_{-1}}{\Psi_{-1}}$$

If the number of samples is large enough, the last step is correct since $r'/r \approx \mathbb{E}[r'/r] = D'/D$ and since $\mathbb{E}[\Psi'_{-1}] \approx \Psi'_{-1}$ and $\mathbb{E}[\Psi_{-1}] \approx \Psi_{-1}$. Under most conditions, the ratio estimator requires only a constant number of samples to converge. Thus, the main computational effort is in estimating n, which is surprisingly lower than that of directly estimating n'. To see this, let r_c and r'_c be the number of samples needed to estimate the sizes of the graph and the subgraph respectively. Since, in a random walk we only hit a node in the subgraph with probability D'/D, we are expected to require Dr'_c/D' random samples to obtain r'_c samples from the subgraph. Thus, if $r'_c/D' \geq r_c/D$ the second method is preferable. Note that this holds in the natural situation that the nodes' degree distributions of the graph and subgraph are similar.

4. NON-UNIQUE ELEMENT COUNTING ESTIMATOR

In this section we present another estimator which is based on counting non-unique elements instead of collisions. On the one hand, it tends to be consistently, yet marginally, more accurate. On the other hand, its proof is much more involved. We thus choose to present the estimator along with its performance (in the experimental results section) without providing a proof of its correctness.

An element in the sample is considered non-unique if it was sampled at least once before. This is slightly different from counting collisions. For example, in the sequence $\{1, 2, 3, 1, 1\}$ there are two non-unique elements (the last two 1's) but three collisions ($x_1 = x_4$, $x_1 = x_5$, and $x_4 = x_5$). The intuition is that counting non-unique elements is less sensitive to errors in which a specific node is oversampled. This is because the non-unique count is linear in the number of times each item was sampled whereas the collision count is quadratic.

We estimate n by $\tilde{n}$ which is the unique solution to the following fixed point equation:

$$\tilde{n} = r - \tilde{C} + \frac{\tilde{n}}{\Psi_{-1}}\sum_{i=1}^{r}\frac{1}{d_{x_i}}\left(1 - \frac{d_{x_i}\Psi_{-1}}{\tilde{n}r}\right)^r \qquad (2)$$

Note that r, $\tilde{C}$, Ψ_{-1} and d_{x_i} are all observed quantities. To see why this is correct, first note that the expected number of non-unique elements is $\mathbb{E}[\tilde{C}] = r - n + \sum_{i=1}^{n}(1-p_i)^r$. Now, consider that

$$\sum_{i=1}^{n}(1-p_i)^r = \mathbb{E}[\frac{1}{p_i}(1-p_i)^r] \approx \frac{1}{r}\sum_{i=1}^{r}\frac{D}{d_{x_i}}(1-\frac{d_{x_i}}{D})^r \ .$$

Also, $D \approx \frac{rn}{\Psi_{-1}}$. Making these substitutions into the expectation expression gives the above fixed point equation. Intuitively, the size estimate $\tilde{n}$ is chosen such that the observed number of non-unique elements is equal to its expectation. As a remark, if the node distribution is uniform, this estimator is identical to the maximum likelihood estimator [7].

5. EXPERIMENTAL EVALUATION

5.1 Networks of known sizes

In order to test our estimators' accuracy we first experimented with three networks whose exact sizes are known.

A synthetic network; a synthetically constructed graph consisting of 1 million nodes whose degree distribution is Zipfian with parameter $\alpha = 2$ and maximal degree $d_m = 1,000$.

The Digital Bibliography and Library Project; we used the Digital Bibliography and Library Project's (DBLP) entire database.[8] Edges in the graph were associated with co-authorship of at least one paper. The resulting graph included 845,211 nodes, each with at least one edge (authors with no co-authors were omitted).

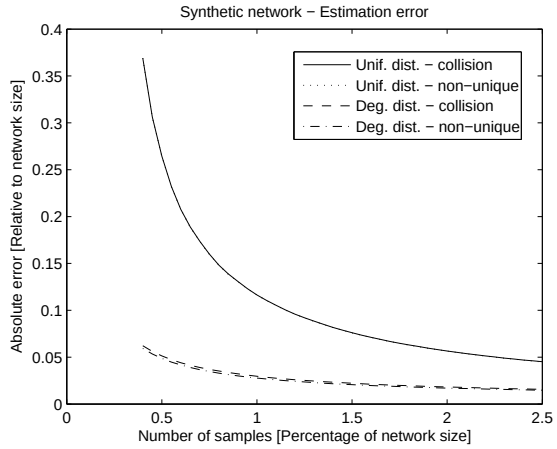
The Internet Movie Database; we useed public Internet Movie Database's (IMDB) entire database.[9] Edge connections between actors were established according to joint participation in at least one movie or TV episode. The resulting graph included 1,955,508 nodes.

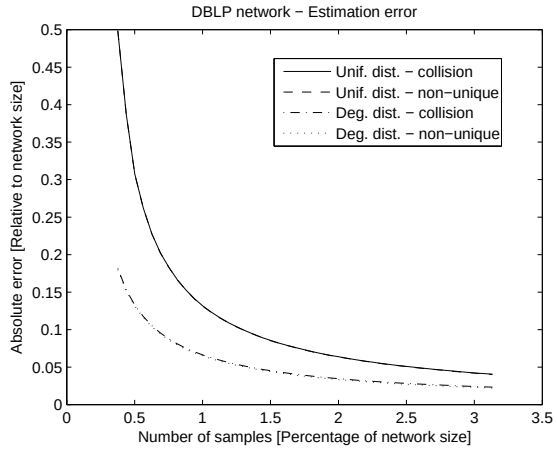
We produced three types of curves. All three were plotted as functions of the percent of sampled nodes. *Error curves* present the normalized absolute estimation error, i.e., $|n - \hat{n}|/n$ where n is the true size of the network and $\hat{n}$ is our estimate of it. *Confidence interval curves* give for each estimator the 5'th and 95'th percentile value from 10,000 independent estimations. In other words, 90% of the estimated sizes fell between the lower and upper curves. Lastly, *comparison curves* present the ratio between the errors of the non-unique element estimator and that of the collision estimator. It is important to stress that this ratio is between the normalized absolute estimation errors and *not* between the estimated values. All presented plots were produced by averaging over 10,000 independent experiments.

Examining the *error curves* depicted in Figure 1, the superiority of degree sampling estimation (both non-unique and collisions based) over uniform sampling estimation is well observed. In particular, for the synthetic network, uniform sampling estimation requires 5 times as many samples as required by degree sampling estimation (0.5% vs. 2.5%), to ensure a normalized absolute estimation error of less than 5%. Also, for the IMDB network, uniform sampling estimation required almost 3 times more samples than required by degree sampling estimation (0.3% vs. 0.8%) to ensure a normalized absolute error of less than 10%.

[8]The DBLP database can be found at http://dblp.uni-trier.de/xml/.

[9]The IMDB database can be found at ftp://ftp.fu-berlin.de/pub/misc/movies/database/.





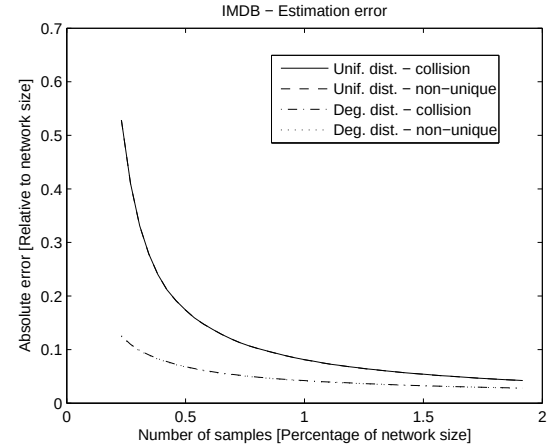
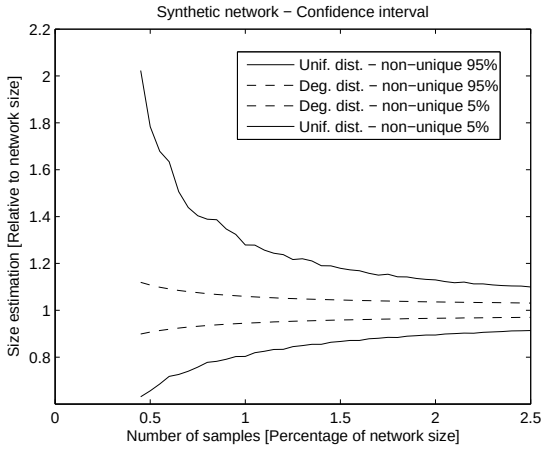
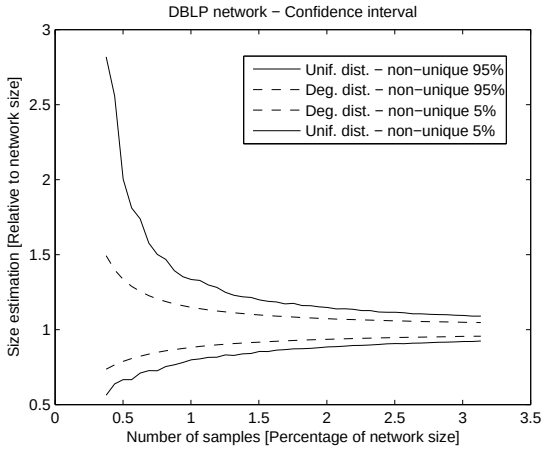


Figure 1: *Error curves* - absolute normalized size estimation errors vs. the percent of sampled nodes for three networks: a 1-million-node synthetic network (top), a network constructed from the Digital Bibliography and Library Project (DBLP) database (middle), and a network obtained from the Internet Movie Data Base (IMDB) database (bottom).





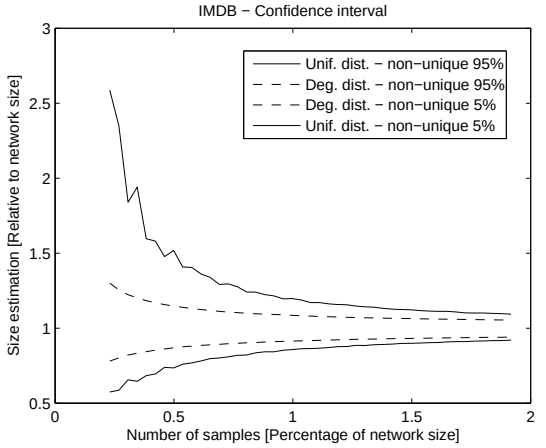


Figure 2: *Confidence interval curves* - relative estimated size vs. the percent of sampled nodes for three networks: a 1-million-node synthetic network (top), a network constructed from the Digital Bibliography and Library Project (DBLP) database (middle), and a network obtained from the Internet Movie Data Base (IMDB) database (bottom).

Similar observations regarding the estimation error are also notable examining the *confidence interval curves* depicted in Figure 2. These curves also demonstrate that there is an inherent asymmetrical bias towards size overestimation. This is probably because both estimators are inversely proportional to the number of collisions or non-

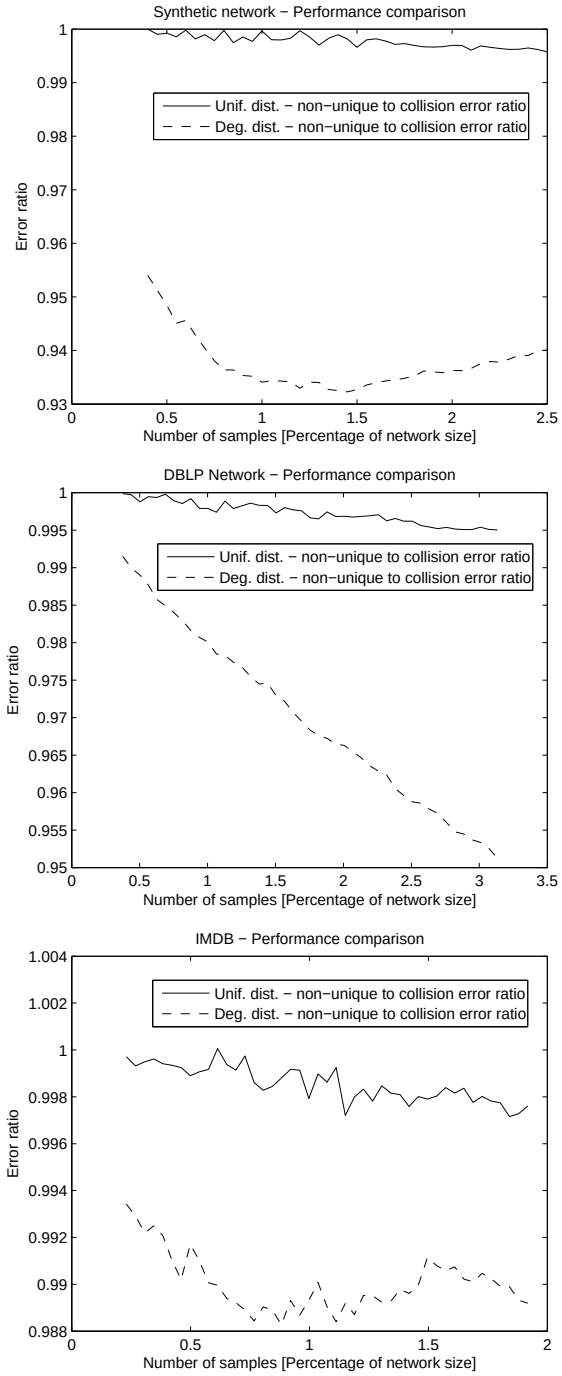


Figure 3: *Comparison curves* - absolute relative error ratio between the non-unique and collision estimators vs. the percent of sampled nodes for three networks: a 1-million-node synthetic network (top), a network constructed from the Digital Bibliography and Library Project (DBLP) database (middle), and a network obtained from the Internet Movie Data Base (IMDB) database (bottom).

unique elements. For example, a 50% discrepancy between the observed number of collisions and its expectation can result in 100% size overestimation but only in 35% size underestimation.

In all the curves above and for all three networks the non-unique elements estimator slightly outperformed the collisions based estimator. This phenomenon is visible when examining the *comparison curves* depicted in Figure 3. For example, for DBLP, the non-unique elements estimator provides a 5% reduced relative error over the collision based estimator when 3% of the network is sampled. The reader should note that the actual size estimates in this case differ by only 0.25%.

In all the aforementioned experiments we estimated the sizes of networks (whose sizes were already known) up to precision of a few single percents. We observed that both collision and non-unique based estimators perform well and that degree based sampling is significantly preferred to uniform sampling. In the next section we estimate the size of a subnetwork within Facebook and size of their entire network.

5.2 Facebook

We used two crawls performed on Facebook by the Authors of [8]. The first crawl consisted of 984,830 uniformly sampled users collected during April 2009.[10] The second crawl was performed during October 2010 and consisted of 988,116 users. This crawl performed a simple random walk on the Facebook graph and therefore selected users with probability proportional to their degree.

5.2.1 Subnetwork size estimation

Since the actual size of Facebook is not known (other than Facebook's own reports) we first estimate the size of a subgraph whose size is known. We selected a random subset of 1,000,000 Facebook users and tried to estimate the size of this sub-population using the first algorithm in Section 3.2. This is done for two reasons. First, to test the subgraph size estimation algorithm. Second, to make sure that Facebook's network's topology and statistics are suitable for our estismators. We present an *error curve*, a *confidence interval curve*, and a *comparison curve* in Figure 4. Note that the x-axis here gives the percent of nodes sampled from the subnetwork and not the entire network as before.

These results corroborate that our subgraph size estimators behave almost identically to the complete graph estimators. This was expected since their analysis is essentially identical. A more important discovery is that the network topology and node degree distribution of Facebook are indeed suitable for our estimators to perform well.

5.2.2 Estimating the size of Facebook

We now estimate the size of the entire Facebook network. Presenting accuracy plots in this case is not possible since the true size of Facebook is not known. The uniform Facebook sample collected during April 2009, contains 2053 collisions and 2052 non-unique elements. Substituting these into Equations (1) and (2) yields estimates of 237,197,785 and 236,984,623 users respectively. The very same month, Facebook ([2]) reported of having "more than 200 million active users" and "more than 250 million active users" three months later. The crawl that was performed during October 2010 contained 4099 collisions and 4064 non-unique users. This gives estimates of 475,566,857 and 475,864,724

[10]The Facebook uniformly sampled crawl can be found at http://odysseas.calit2.uci.edu/research/.

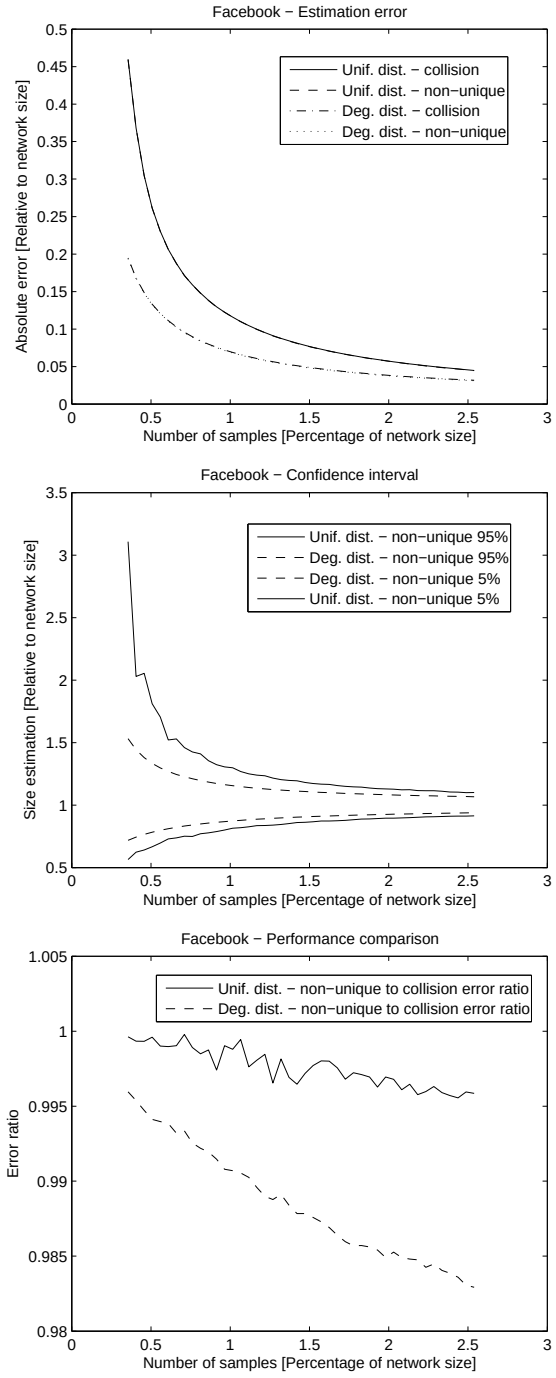


Figure 4: Absolute relative estimation error (top), confidence intervals (middle), and estimation relative error ratio (bottom) vs. the percentage of samples taken from a 1 million user subnetwork of Facebook.

respectively. Facebook at the same time reported of having "more than 500 million active users". This is summarized in Table 1.

Notice that the second crawl sampled half as many users relative to the network size at the time it was made. Yet, it produced roughly twice as many collisions. This is because degree proportional sampling is expected to generate more collisions than uniform sampling. The discrepancy between our estimates and the official reports by Facebook stems from two main reasons. On the one hand, the crawler cannot distinguish between active and non active users. Thus our estimates include non-active users which causes an over estimation. On the other hand, our crawler cannot pass through users whose privacy settings hide their list of friends, which causes underestimation. Thus, our estimates and Facebook's reports give slightly different figures. While Facebook counts "active" users, we estimate the number of Facebook users whose list of friends is visible to the crawler regardless of their activity.

Table 1: Crawl details and consequent size estimates of the entire Facebook network for April 2009 and October 2010.

	April 2009	October 2010
Sampling distribution	uniform	degree
Number of samples	$0.98 \cdot 10^6$	$1 \cdot 10^6$
Number of collisions	2053	4099
Number of non-unique	2052	4064
Collision estimator	$237 \cdot 10^6$	$475 \cdot 10^6$
Nun-unique estimator	$236 \cdot 10^6$	$475 \cdot 10^6$
Facebook report	$200 - 250 \cdot 10^6$	$500 \cdot 10^6$

Since the crawler has no indication of users' activity and since it is unclear what Facebook defines as "active", we cannot offset this effect. However, we can try to estimate the number of blocked users (those whose privacy settings block the crawler). This can be approximated using the fraction of such users in other users' friends lists which yields an estimate of $650 \cdot 10^5$ users, active and non-active.

5.3 Synthetic network - large sample region

Interestingly, for a large enough number of samples (e.g., 30% of the network's size), uniform sampling estimation outperforms degree sampling estimation. This phenomenon repeats itself for all three known size network we examined. We provide an *error curve*, a *confidence interval curve*, and a *comparison curve* in Figure 5 for the synthetic network only but this time extending the number of samples all the way to 100%.

6. CONCLUSIONS

We presented two algorithms for estimating the size of graphs. Both algorithms rely on nodes being samples from the graph's stationary distribution. We showed both analytically and experimentally that, for social-networks and other small world graphs, these algorithms considerably outperform uniformly sampling nodes. They consistently provide more accurate estimates while using a smaller number of samples. This result is even more outstanding since uniformly sampling nodes is strictly harder than sampling them according to the stationary distribution.

However, there is still room for improvement. While performing the random walk in order to sample nodes from the stationary distribution, we encounter many nodes whose degrees or collisions we never use. Using this information can possibly reduce the number of random walk steps even fur-

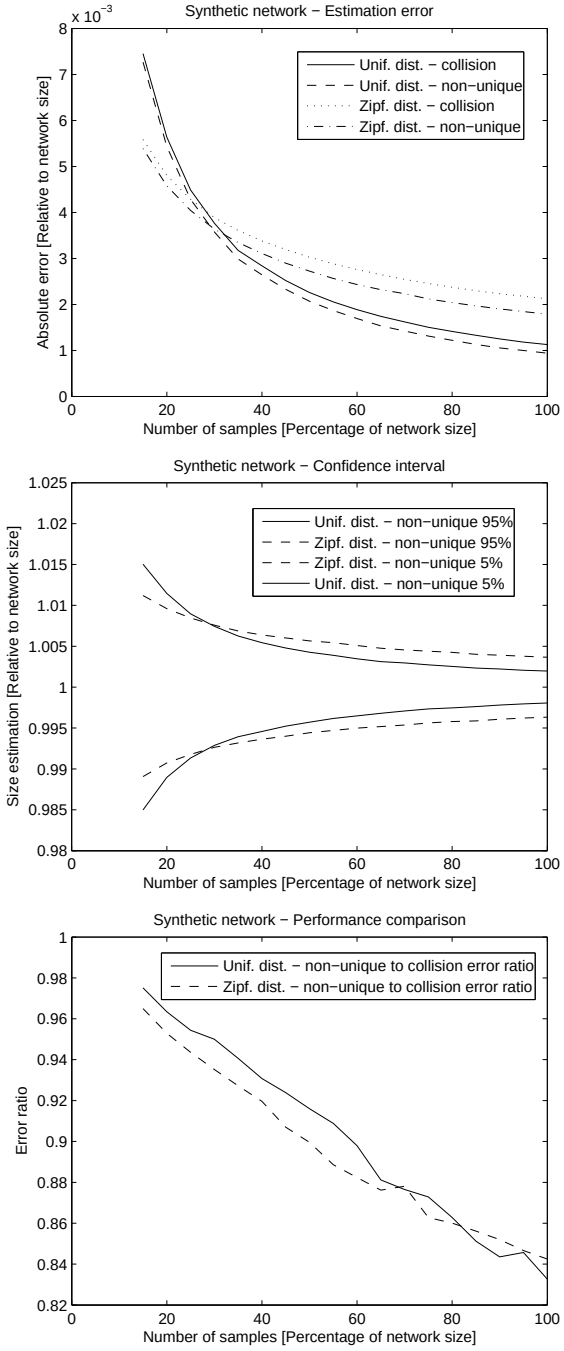


Figure 5: Absolute relative estimation error (top), confidence intervals (middle), and estimation relative error ratio (bottom) vs. the percent of sampled nodes for a 1 million node synthetic network whose node degree distribution is Zipfian. Note the large number of samples relative to the network size.

ther. However, analyzing the entire random walk is hard since this introduces dependencies on the graph topology in a non-trivial way.

7. ACKNOWLEDGMENT

We thank Minas Gjoka for being extremely helpful and providing the Facebook crawls used in the simulations. We also thank Ronny Lampel, Yoelle Maarek and Ravi Kumar for useful discussions.

8. REFERENCES

[1] Estimating the size of a population. http://www.rsscse.org.uk/ts/gtb/johnson.pdf.

[2] Facebook statistics. http://www.facebook.com/press/info.php?statistics.

[3] Z. Bar-Yossef and M. Gurevich. Efficient search engine measurements. In *Proceedings of the 16th international conference on World Wide Web (WWW'07)*, pages 401–410, Banff, Alberta, Canada, 2007.

[4] Z. Bar-Yossef and M. Gurevich. Random sampling from a search engine's index. *J. ACM*, 55(5), 2008.

[5] K. Bharat and A. Broder. A technique for measuring the relative size and overlap of public web search engines. *Comput. Netw. ISDN Syst.*, 30(1-7):379–388, 1998.

[6] A. Chao. Estimating the population size for capture-recapture data with unequal catchability. In *Biometrics*, December 1987.

[7] M. Finkelstein, H. G. Tucker, and J. A. Veeh. Confidence intervals for the number of unseen types. *Statistics & Probability Letters*, 37(4):423–430, March 1998.

[8] M. Gjoka, M. Kurant, C. T. Butts, and A. Markopoulou. Walking in Facebook: A Case Study of Unbiased Sampling of OSNs. In *Proceedings of IEEE INFOCOM '10*, San Diego, CA, March 2010.

[9] S. Han Hee, C. Tae Won, D. Vacha, Z. Yin, and Q. Lili. Scalable proximity estimation and link prediction in online social networks. In *Proceedings of the 9th ACM SIGCOMM conference on Internet Measurement (IMC'09)*, pages 322–335, Chicago, Illinois, USA, 2009.

[10] R. Huggins, H.-C. Yang, A. Chao, and P. S. F. Yip. Population size estimation using local sample coverage for open populations. *Journal of Statistical Planning and Inference*, 113(2):699 – 714, 2003.

[11] D. E. Knuth. Estimating the efficiency of backtrack programs. Technical report, Stanford, CA, USA, 1974.

[12] Y. Koren, S. C. North, and C. Volinsky. Measuring and extracting proximity in networks. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data mining (KDD'06)*, pages 245–255, Philadelphia, PA, USA, 2006.

[13] L. Lovasz. Random walks on graphs. a survey. *Combinatorics*, 1993.

[14] A. Marchetti-Spaccamela. On the estimate of the size of a directed graph. In J. van Leeuwen, editor, *Graph-Theoretic Concepts in Computer Science*, volume 344 of *Lecture Notes in Computer Science*, pages 317–326. Springer Berlin / Heidelberg, 1989.

[15] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. In *Proceedings of the 5th ACM Conference on Internet Measurement (IMC'07)*, San Diego, CA, USA, 2007.

[16] R. Motwani and S. Vassilvitskii. Distinct values estimators for power law distributions. In *Proceedings*

of the Third Workshop on Analytic Algorithmics and Combinatorics (ANALCO'06), Miami , Florida, 2006.

[17] L. Pitt. A note on extending knuth's tree estimator to directed acyclic graphs. *Inf. Process. Lett.*, 24(3):203–206, 1987.

[18] S. Ye and F. Wu. Estimating the size of online social networks. In *Proceedings of the IEEE Second International Conference on Social Computing (SocialCom)*, pages 169–176, Aug. 2010.

[19] A. Yong-Yeol, H. Seungyeop, K. Haewoon, M. Sue, and J. Hawoong. Analysis of topological characteristics of huge online social networking services. In *Proceedings of the 16th international conference on World Wide Web (WWW'07)*, pages 835–844, Banff, Alberta, Canada, 2007.

APPENDIX

A. CONCENTRATION OF C, Ψ_1, AND Ψ_{-1}

In the proof of Corollary 1 we required that each of the variables C, Ψ_1, and Ψ_{-1} give an $\epsilon/4$ appropriation their respective expected values with probability at least $1-\delta/3$. From Chebyshev's inequality we get that this is true for any variable Z for which

$$\mathrm{Var}[Z] \leq \delta\epsilon^2 \mathbb{E}^{\,2}[Z]/48. \tag{3}$$

To use this we must first bound the variance of all three variables.

We start with C, the number of collisions. We remind the reader our notations. $Y_{i,j} = 1$ if $x_i = x_j$ and 0 else, where $\{x_1 \ldots, x_r\}$ are the r sampled nodes. Moreover, $C = \sum_{i<j}^{r} Y_{i,j}$. We compute $\mathbb{E}\,[C^2]$ using the linearity of the expectation.

$$\mathbb{E}\,[C^2] = \mathbb{E}\,[(\sum_{i<j}^{r} Y_{i,j})^2] = \sum_{i<j}^{r}\sum_{i'<j'}^{r} \mathbb{E}\,[Y_{i,j}Y_{i',j'}]$$

To calculate the last summation we divide it into three cases. When $i = i'$ and $j = j'$ we have that $\mathbb{E}\,[Y_{i,j}Y_{i',j'}] = \mathbb{E}\,[Y_{i,j}] = \sum_{\ell=1}^{n} p_\ell^2$. Moreover, we have $\binom{r}{2}$ such case. In the case where $i \neq i'$ and $j \neq j'$, the number of the summands is equal to the number of ways one chooses 4 elements out of r elements and then split them to two pairs, i.e., $6\binom{r}{4}$. Also, in this case $\mathbb{E}\,[Y_{i,j}Y_{i',j'}] = (\sum_{\ell=1}^{n} p_\ell^2)^2$. In third and last possibility is where $i = i'$ and $j \neq j'$ or $i \neq i'$ and $j = j'$. We have $3!\binom{r}{3}$ such summands because this is the number of ways to choose 3 elements out of r and then decide for each index if it is from the lest sum, right sum, or both. In this case we have $\mathbb{E}\,[Y_{i,j}Y_{i,j'}] = \sum_{\ell=1}^{n} p_\ell^3$. Summing those up we get that:

$$\begin{aligned}
\mathrm{Var}[C] &= \mathbb{E}\,[C^2] - \mathbb{E}^{\,2}[C] \\
&= \binom{r}{2}\sum_{\ell=1}^{n} p_\ell^2 + 6\binom{r}{3}\sum_{\ell=1}^{n} p_\ell^3 \\
&\quad + 6\binom{r}{4}(\sum_{\ell=1}^{n} p_\ell^2)^2 - \binom{r}{2}^2(\sum_{\ell=1}^{n} p_\ell^2)^2 \\
&\leq \binom{r}{2}\sum_{\ell=1}^{n} p_\ell^2 + 6\binom{r}{3}\sum_{\ell=1}^{n} p_\ell^3
\end{aligned}$$

Substituting this into Equation 3 and demanding that each of the summoned is less than half the right hand side gives us the condition for r_c:

$$r_c \geq \max\left\{\frac{8\sqrt{3}}{\epsilon\sqrt{\delta}\sqrt{\sum_{i=1}^{n} p_i^2}}, \frac{384\sum_{i=1}^{n} p_i^3}{\epsilon^2\delta(\sum_{i=1}^{n} p_i^2)^2}\right\}$$

Since the nodes are picked independently, computing the variance of Ψ_1 is easy:

$$\mathrm{Var}[\Psi_1] = r\,\mathrm{Var}[d_{x_1}] < r\mathbb{E}\,[d_{x_1}^2] = r\sum_{i=1}^{n} p_i d_i^2 = rD^2\sum_{i=1}^{n} p_i^3$$

From Equation 3, and the expectation of Ψ_1 calculated in the body of the paper we get the requirement:

$$r_c > \frac{48\sum_{i=1}^{n} p_i^3}{\epsilon^2\delta\left(\sum_{i=1}^{n} p_i^2\right)^2}$$

We do the same for Ψ_{-1}:

$$\begin{aligned}
\mathrm{Var}[\Psi_{-1}] &= r\,\mathrm{Var}[1/d_{x_1}] < r\mathbb{E}\,[1/d_{x_1}^2] \\
&= r\sum_{i=1}^{n} p_i/d_i^2 = \frac{r}{D^2}\sum_{i=1}^{n}\frac{1}{p_i}
\end{aligned}$$

Thus Ψ_{-1} is also $\epsilon/4$ close to its expectation with probability at least $1-\delta/3$ if: which is satisfied by

$$r_c \geq \frac{48\sum_{i=1}^{n} 1/p_i}{\epsilon^2\delta n^2}$$

Combining them gives us that C, Ψ_1 and Ψ_{-1} are individually $\epsilon/4$ close to their expectation, each with probability at least $1-\delta/3$ if:

$$r_c \geq \max\left\{\frac{8\sqrt{3}}{\epsilon\sqrt{\delta}\sqrt{\sum_{i=1}^{n} p_i^2}}, \frac{384\sum_{i=1}^{n} p_i^3}{\epsilon^2\delta(\sum_{i=1}^{n} p_i^2)^2}, \frac{48\sum_{i=1}^{n} 1/p_i}{\epsilon^2\delta n^2}\right\}$$

Note that the condition for the concentration of Ψ_1 is subsumed by the conditions for the concentration of C.

B. PARTIAL LISTS OF FRIENDS

In some settings, only a random subset of the list of friends is given to the crawler. Thus, assuming the subset of given friends is truly random, the crawler can keep performing the random walk undisturbed. However, it can no longer evaluate Ψ_1 and Ψ_{-1}. To this end, we provide estimators that do not require Ψ_1 and Ψ_{-1}. They do, however, require a priori knowledge of nodes' degree distribution. A similar assumption was also made in [16] where the authors presented an algorithm for estimating the number of attributes in a database.

Defining as before the ℓ'th moment of the degree distribution as $\mathcal{M}_\ell = \mathbb{E}\,[d^\ell]$ the collision estimator becomes

$$\hat{n} = \binom{r}{2}\frac{\mathcal{M}_2}{C\mathcal{M}_1^2}\ .$$

For the non-unique estimator we get

$$\hat{n} = r - C + \sum_{i=1}^{d_m}\hat{n}P[d=i]\left(1 - \frac{i}{\hat{n}\mathcal{M}_1}\right)^r\ .$$