A Distributed Algorithm for Identifying Information Hubs in Social Networks

Muhammad U. Ilyas, *Member, IEEE*, M. Zubair Shafiq, *Student Member, IEEE*, Alex X. Liu, *Member, IEEE*, Hayder Radha, *Fellow, IEEE*

***Abstract*—This paper addresses the problem of identifying the top-k information hubs in a social network. Identifying top-k information hubs is crucial for many applications such as advertising in social networks where advertisers are interested in identifying hubs to whom free samples can be given. Existing solutions are centralized and require time stamped information about pair-wise user interactions and can only be used by social network owners as only they have access to such data. Existing distributed algorithms suffer from poor accuracy. In this paper, we propose a new algorithm to identify information hubs that preserves user privacy. Our method can identify hubs without requiring a central entity to access the complete friendship graph. We achieve this by fully distributing the computation using the Kempe-McSherry algorithm, while addressing user privacy concerns. We evaluate the effectiveness of our proposed technique using three real-world data set; The first two are Facebook data sets containing about 6 million users and more than 40 million friendship links. The third data set is from Twitter and comprises of a little over 2 million users. The results of our analysis show that our algorithm is up to 50% more accurate than existing algorithms. Results also show that the proposed algorithm can estimate the rank of the top-k information hubs users more accurately than existing approaches.**

***Index Terms*—Social network analysis; spectral graph analysis; eigendecomposition; information hubs.**

I. Introduction

A. Background and Motivation

In a social network, a user that has a large number of interactions with other users is defined as an *information hub* (or simply a *hub*) [9]. An interaction refers to the transmission of information by one user to another user. For example, an interaction from user A to user B in online social networks may be the action when user A posts a message or comment on user B's profile. Hubs play important roles in the spread or subversion of propaganda, ideologies, or gossips in social networks. Taking the advertising industry as an example, instead of giving free product samples to random people, to improve the effectiveness of word of mouth advertising and increase recommendation based product adoption, they may want to give free samples to hubs only [14]. For example, CNN reported that Samsung used social networks information to target dissatisfied owners of Apple iPhone 4 in a recent advertisement campaign [13]. Samsung first monitored Twitter feeds to identify dissatisfied iPhone 4 owners who are the most active in terms of communicating with their friends (*i.e.* hubs) and are therefore most influential in spreading word of mouth recommendation, then offered free GalaxyS phones to some of them. Furthermore, observing adoption of products or trends at hubs helps to predict the eventual total sale of a product [14]. For instance, advertisers can observe the impact of distributing free samples to hubs to predict the future successfulness of a product. Due to limited advertisement budget (*e.g.*, free product samples), advertisers want to identify the top-k nodes in a social network. Suri and Narahari [33] defined these as, "for any given positive integer k, the top-k nodes in a social network, based on a certain measure appropriate for the social network." In the context of this research, that measure is information. Therefore, identifying top-k information hubs in social networks is an important problem.

B. Limitations of Prior Art

Prior methods for computing top-k information hubs (*e.g.*, [26] and [15]), are mostly centralized assuming the availability of either interaction or friendship graphs. The interaction graph of a social network is a directed multigraph [5] whose nodes represent users and directed links represent the existence of a directed pair-wise interaction. Each link is labeled with a time stamp that indicates when the interaction occurred. The friendship graph of a social network consists of nodes representing users and undirected links representing the friend relationship among users. Figure 1 shows the conceptual depiction of the friendship graph between users and the overlaid interaction graph. However, centralized computation of top-k information hubs is mostly unrealistic for parties such as advertisers because online social networking companies are reluctant to share their interaction or friendship graphs due to privacy concerns and regulations [1]. Furthermore, advertisers cannot even directly collect interaction or friendship information from

Manuscript received February 15, 2012; revised July 23, 2012 and October 29, 2012.

Muhammad U. Ilyas is now with the Department of Electrical Engineering, School of Electrical Engineering & Computer Science, National University of Sciences & Technology, Islamabad - 44000, Pakistan, but most of this work was conducted while he was a post-doctoral researcher in both the Department of Computer Science and Engineering and the Department of Electrical and Computer Engineering, Michigan State University, East Lansing, Michigan - 48824 U.S.A. Email: usman.ilyas@seecs.edu.pk.

M. Zubair Shafiq and Alex X. Liu are with the Department of Computer Science and Engineering, Michigan State University, East Lansing, Michigan - 48824 U.S.A. Email: {shafiqmu, alexliu}@cse.msu.edu, Tel: +1-517-353-5152, Fax: +1-517-432-1061.

Hayder Radha is with the Department of Electrical and Computer Engineering, Michigan State University, East Lansing, Michigan - 48824. Email: radha@egr.msu.edu.

This work was supported in part by the National Science Foundation under Grant IIS-0968495.

A preliminary version of this paper titled "A Distributed and Privacy-Preserving Algorithm for Identifying Information Hubs in Social Networks" was published in Proceedings of the 30th Annual IEEE Conference on Computer Communications (INFOCOM) Mini-Conference, pages 561-565, Shanghai, China, April 2011.

A. X. Liu is the corresponding author of this paper.

social network sites by means such as crawling because for many online social networking companies such as Facebook [3], unauthorized data collection is a violation of their terms of service.

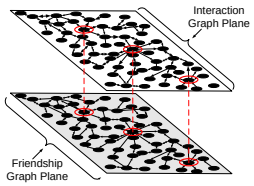


Fig. 1: Conceptual depiction of the friendship graph between users and the overlaid interaction graph.

C. Proposed Solution

In this paper, we propose a distributed and privacy preserving algorithm for computing top-k information hubs in social networks. Distributed algorithms for computing top-k information hubs have to be privacy preserving because users are typically hesitant to disclose explicit information about their friendship links or interaction information due to privacy concerns [35]. To preserve the privacy of user interactions, our algorithm is distributed and does not require the advertiser to know users' friendship associations or interactions. There are three technical challenges in designing such an algorithm. First, the problem of inferring a user's salience (whose ground truth resides in the interaction graph) from the corresponding friendship graph is inherently difficult because an interaction graph has more information than its corresponding friendship graph. Furthermore, a friendship graph is directed or undirected (and un-weighted), whereas the interaction graph is a directed multigraph. Second, the complete friendship graph itself may not be available to the parties interested in identifying hubs. Third, preserving users' privacy in this computation is difficult as any information exchange involved in this computation should not contain any personal information.

We now present an overview of our proposed solutions to the above-mentioned technical challenges. To address the first challenge, we apply principal component centrality (PCC), a new measure of centrality we introduced in [17], to the friendship graphs. The intuition behind PCC is that a user who is connected to well-connected users (even if the user himself is poorly connected) has a more central status. For example, a poorly connected person who has a direct connection with a well-connected representative in some population may be capable of propagating an opinion well by simply convincing the representative. Unlike other measures of user influence (*e.g.*, eigenvector centrality [6], [7]), PCC takes into consideration the fact that social networks can be multi-polar consisting of multiple communities with few connections between them. However, since PCC applies to symmetric matrices, it does not apply to social graphs in which relationships are not bi-directional. We extend PCC's definition to the Generalized PCC which is applicable to non-symmetric matrices as well, i.e. it is applicable to social graphs in which relationships are directional.

To address the second challenge of friendship graph data availability, we distribute the computation of PCC among users and therefore do not require a central entity to access the friendship graph. Advertisers can utilize existing functionality in popular online social networks (such as *groups* and *pages* in Facebook [2]) to implement our proposed distributed method. Motivation for user participation in decentralized PCC computation may range from tangible incentives such as receiving free samples from advertisers (*e.g.* [13]), to intangible incentives such as bragging rights about one's popularity (*e.g.* [31]). We decentralize the PCC computation using the Kempe-McSherry (KM) algorithm [18]. These iterative algorithms compute eigenvalues and eigenvectors that are essential for computing nodes' PCCs. Our decentralized algorithm restricts the set of users that a particular user has to communicate with to its immediate friends. Furthermore, the memory requirement at each user of this algorithm grows only linearly with the number of friends. Hence, one of the contributions of this work is extending the original centralized PCC approach to a more practical distributed PCC form. This new distributed PCC form is an accurate and robust centrality measure that is capable of identifying all salient users in a social graph using a truly decentralized and scalable method. Finally only a numeric vector representing intermediate eigenvector entries are exchanged between users. It is impossible to reverse-engineer users' friendship associations from these intermediate scores.

D. Results and Findings

We evaluated the effectiveness of our proposed technique using three real-world data sets: The first two are Facebook data sets [34] containing about 6 million users and more than 40 million friendship links. In this data set relationships are reciprocated, i.e. links between users are undirected (if user A is friends with user B, that implies user B is also friends with user A). The third data set consists of a little over 2 million Twitter users and their activity. The Twitter data set differs from the Facebook data sets in that relationships between users are not necessarily reciprocated, i.e. links between users are directed (if user A follows user B, that does not imply that user B also follows user A). We have four major findings from our results. First, there is indeed close correlation between the PCC of nodes in the friendship graph and corresponding dynamic user interaction data. We envision that this correlation can be exploited for other purposes as well. Second, the computation of PCC can be effectively distributed across individual users in a social graph without compromising its accuracy. This eliminates the requirement of a central authority for identifying hubs. Third, the accuracy of PCC improves as we use more eigenvectors in its computation. Further, the appropriate number of eigenvectors required in the computation of PCC for real-world social networks is around 10-20. Fourth, the accuracy (in terms of number of correctly identified top-k users and their estimated rank) of PCC improves as the duration of interaction data used for comparison is increased from 1 month, to 6 month to more than a year. This essentially shows that PCC scores reflect the flow of information between users of a social network over long time periods.

E. Key Contributions

We make four key contributions in this paper. First, we propose a novel method to infer information lying in the interaction graph (*i.e.* hub identification) from the friendship graph in social networks without using the interaction data. Earlier works are limited to solving this problem using complete interaction graph data. Second, our proposed method, first of its kind, allows third parties (other than social network owners) to solve this problem. We use a distributed method to overcome the requirement of a central authority. Third, our proposed method preserves the privacy of users, *i.e.*, users do not release any personal information to other users. We achieve this objective by letting each user share only some real numbers that cannot be reverse-engineered. Finally, we evaluate the effectiveness of our proposed technique using real-world Facebook data sets that are publicly available. The results of our analysis show that the proposed approach improves the accuracy of finding the top-k user set by approximately 50% over existing measures. Furthermore, the proposed technique accurately estimates the rank of individual users.

The rest of the paper proceeds as follows. In Section II, we present an overview of related work. We provide the details of our proposed approach in Section III. We also provide the analysis of the data set used for evaluating our proposed technique in Section IV-A. We then provide the detailed results of our evaluation in the rest of Section IV. Finally, we conclude the paper in Section V.

II. Related Work

Besides work on hub identification using user interaction data ([15], [26]) mentioned in Section I, we provide an overview of other research on influence maximization, a different but related problem. Several algorithms have been proposed for identifying influential users in social networks [12], [19]–[21], [33], [36]. The objective function for this influence maximization problem is to maximize the number of users that are part of information flows initiated by the top-k influential users. In contrast, our method uses friendship graphs, is fully distributed, and is privacy-preserving, while such work uses user interaction data and is centralized and is not privacy-preserving.

Kempe *et al.* studied this influence maximization problem for the first time, proved it is NP-hard, and proposed a heuristics-based algorithm that achieves 63% of the optimal result in most cases and outperforms degree and distance centrality heuristics [19]. Suri and Narahari later proposed Shapley value based heuristic for solving this problem [33]. Zou *et al.* studied the same problem with an additional constraint of latency [36]. Estevez *et al.* proposed an algorithm called the Set Covering Greedy (SCG) algorithm, which takes into account the intuition that we should prefer to select nodes in different neighborhoods rather than selecting highly connected nodes lying in the same neighborhood [12]. Kimura *et al.* studied the influence maximization problem with respect to two widely-used fundamental stochastic information diffusion models in networks, and proposed a solution utilizing tools from bond percolation and graph theory [20], [21].

Algorithms that forgo using interaction data use structural information like the friendship graph. They are based on centrality measures computed from friendship graph topologies. Marsden [27] used degree, closeness, betweenness and eigenvector centrality measures. This is followed by Shi *et al.* in [32] who used the same centrality measures, *i.e.* degree, closeness, betweenness and pagerank [22], [24], [30] (which is just an iterative algorithm to compute eigenvector centrality). However, as Borgatti showed in [8], degree, closeness and betweenness centrality are inappropriate measures of centrality for influence processes. Degree centrality is a good measure of the rate of immediate rate of spread of influence from nodes in the short-term. Betweenness and closeness centrality are ill-suited for the problem at hand because the definitions underlying them assume that the flow on the network does not replicate and occurs only along shortest paths. Therefore, the performance of Marsden's use of eigenvector centrality and Shi's use of Pagerank form the baseline for comparison against our proposed algorithm. Canright, Engø-Monsen and Jelasity [10] described a distributed and privacy preserving algorithm for the computation of eigenvector centrality/PageRank.

III. Proposed Solution

This section presents our proposed technique for identifying information hubs in social networks. We model the information flow as an influence process. The underlying rationale for doing so is rooted in the assumption that in social networks people (nodes) with more friends (connections) send and receive more messages. Furthermore, people will receive more messages from friends that send/receive a lot of traffic than from those that send/receive fewer messages. This information flow can be modeled as an influence process. According to Borgatti's two dimensional taxonomy of node centrality measures in [8], the appropriate measure to quantify nodes' influence is eigenvector centrality (EVC) [6], [7].

A. Eigenvector Centrality

Let $\mathbf{A}$ denote the adjacency matrix of a graph $G(V, E)$ consisting of the set of nodes $V = \{v_1, v_2, v_3, \ldots, v_N\}$ of size N and set of undirected edges E. When a link is present between two nodes v_i and v_j, both $A_{i,j}$ and $A_{j,i}$ are set to 1 and set to 0 otherwise. Let $\Gamma(v_i)$ denote the neighborhood of v_i, the set of nodes v_i is connected to directly. EVC of a node is recursively defined as proportional to the number of its neighbors and their respective EVCs. Let $x(i)$ be the EVC score of a node v_i. Then,

$$x(i) = \frac{1}{\lambda_1} \sum_{v_j \in \Gamma(v_i)} x(j) = \frac{1}{\lambda_1} \sum_{j=1}^{N} A_{i,j} x(j) \tag{1}$$

Here λ_1 is a constant (later found to be the principal eigenvalue of $\mathbf{A}$). Equation 1 can be rewritten in vector form Equation 2 where $\mathbf{x}_1 = [x(1), x(2), x(3), \ldots, x(N)]^T$ is the vector of EVC scores of all nodes.

$$\mathbf{x}_1 = \frac{1}{\lambda_1} \mathbf{A}\mathbf{x}_1 \iff \lambda_1 \mathbf{x}_1 = \mathbf{A}\mathbf{x}_1 \tag{2}$$

Equation 2 is the well-known eigenvector equation where this centrality takes its name from. Obviously several eigenvalue/eigenvector pairs exist for an adjacency matrix $\mathbf{A}$. Here,

λ_1 is the largest of all eigenvalues of $\mathbf{A}$ by magnitude. If λ_i is any other eigenvalue of $\mathbf{A}$ then $|\lambda_1| > |\lambda_i|$. The eigenvector $\mathbf{x}_1 = [x_1(1), x_1(2), \ldots, x_1(N)]^T$ corresponding to the principal eigenvalue is the principal eigenvector. Thus, the vector of node EVCs is equivalent to the principal eigenvector. The EVC of a node v_i is the corresponding element $\mathbf{x}_1(i)$ of the principal eigenvector $\mathbf{x}_1$.

B. Motivation for Principal Component Centrality

As we demonstrated in [17], in networks of multiple communities with sparse connectivity between communities, EVC assigns centrality scores to nodes according to their location with respect to the most dominant community. We also gave an elaborate example illustrating the relationship between successive eigenvectors and their interpretation in terms of the network topology of the graph adjacency matrix they are obtained from. When applied to large networks, EVC fails to assign significant scores to a large fraction of nodes. The principal eigenvector is "pulled" in the direction of the largest community. The motivation for using PCC as a measure of node influence may be understood by looking at EVC in the context of principal component analysis (PCA) [11]. In PCA, when feature vectors are extracted from an $N \times N$ covariance matrix of N random variables, the principal eigenvector is the most dominant feature vector, i.e. the direction in N-dimensional hyperspace along which the spread of data points is greatest. Similarly, the second eigenvector (corresponding to the second largest eigenvalue) is representative of the second most significant feature of the data set. The second eigenvector may also be thought of as the most significant feature after the data points are collapsed along the direction of the principal eigenvector. Eigendecomposition of a covariance matrix is the performed using the well-known PCA. PCA is used to compute the eigenvectors $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \ldots, \mathbf{x}_N$ and eigenvalues $\lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_N$ of the graph G's adjacency matrix $\mathbf{A}$. Readers interested in a deeper coverage of the intuitive meaning of eigenvectors and eigenvalues of a graph adjacency matrix are referred to Ilyas and Radha [17].

C. Definition of PCC

While EVC assigns centrality to nodes according to their location with respect to the most dominant community in a graph G, PCC takes into consideration additional communities. We define the PCC of a node in a graph as its Euclidean distance/ℓ^2 norm from the origin in the P-dimensional eigenspace. The basis vectors of that eigenspace are the P most significant eigenvectors of the adjacency matrix A of the graph G under consideration. For a graph G, its N eigenvalues $|\lambda_1| \geq |\lambda_2| \geq \ldots \geq |\lambda_N|$ correspond to the normalized eigenvectors $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$, respectively. The eigenvector/eigenvalue pairs are indexed in descending order of magnitude of eigenvalues. When $P = 1$, PCC equals a scaled version of EVC. The parameter P in PCC can be used as a tuning parameter to adjust the number of eigenvectors included in PCC.

Let $\mathbf{X}$ denote the $N \times N$ matrix of concatenated eigenvectors $\mathbf{X} = [\mathbf{x}_1 \mathbf{x}_2 \ldots \mathbf{x}_N]$ and let $\Lambda = [\lambda_1 \lambda_2 \ldots \lambda_N]^T$ be the vector of eigenvalues. Furthermore, if $P < N$ (typically $P \ll N$) and if matrix $\mathbf{X}$ has dimensions $N \times N$, then $\mathbf{X}_{N \times P}$ will denote the submatrix of $\mathbf{X}$ consisting of the first N rows and first P columns. Then PCC can be expressed in matrix form as:

$$\mathbf{C}_P = \sqrt{((\mathbf{A}\mathbf{X}_{N\times P}) \odot (\mathbf{A}\mathbf{X}_{N\times P}))\, \mathbf{1}_{P\times 1}} \tag{3}$$

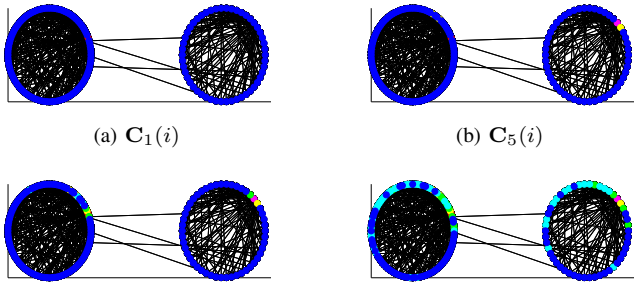
The '$\odot$' operator is the Hadamard (or entrywise product or Schur product) operator and $\mathbf{1}_{P\times 1}$ is a vector of 1s of length P. Equation 3 can also be expressed in terms of the eigenvalue and eigenvector matrices Λ and $\mathbf{X}$, of the adjacency matrix $\mathbf{A}$:

$$\mathbf{C}_P = \sqrt{(\mathbf{X}_{N\times P} \odot \mathbf{X}_{N\times P})\, (\Lambda_{P\times 1} \odot \Lambda_{P\times 1})}. \tag{4}$$

D. Generalized PCC

So far, we have assumed that the adjacency matrix of social networks is symmetric. This limits application of the PCC, as it has been defined in the preceding section, to undirected graphs. However, in several social network services, relationships between users is one-directional and not necessarily reciprocated. The adjacency matrix of such a network is non-symmetric, and with exception of the principal eigenvalue and eigenvector all subsequent eigenvalues and eigenvectors may be complex. To extend the possible use of PCC to social networks whose topology is best captured by directed graphs, we generalize the definition of PCC as follows.

$$\mathbf{C}_P = \sqrt{|(\mathbf{X}_{N\times P} \odot \mathbf{X}_{N\times P})|\, |(\Lambda_{P\times 1} \odot \Lambda_{P\times 1})|}. \tag{5}$$



(a) $\mathbf{C}_1(i)$ (b) $\mathbf{C}_5(i)$ (c) $\mathbf{C}_{10}(i)$ (d) $\mathbf{C}_{100}(i)$

Fig. 2: PCC of nodes in a network consisting of two Barabási-Albert graphs of 100 and 50 nodes connected by a few links when computed using the most significant (a) 1, (b) 5, (c) 10, and (d) 100 eigenvectors. Warm node colors represent high PCC, cold node colors represent low PCC.

Node PCCs as defined in equations 3, 4 and 5 are not normalized. To allow interpretation of centrality scores, Ruhnau advocated in [29] that they should be normalized by either the Euclidean norm (ℓ_2 norm) or the maximum norm (ℓ_∞ *i.e.* the maximum centrality score) of the centrality vector. For the remainder of this paper the PCC vector will be normalized by the ℓ_∞ norm, thereby restricting all entries to the range $[0, 1]$.

We demonstrate PCC on a small-scale example, a graph consisting of two Barabási-Albert graphs [4], one consisting of 100 nodes in one community that is sparsely connected with another Barabási-Albert graph of 50 nodes. Figure 2 demonstrates the effect of changing number of eigenvectors P for PCC $\mathbf{C}_P$ from 1 (Figure 2a) for EVC to 5 (Figure 2b), 10 (Figure 2c) and 100 (Figure 2d). As this example shows, EVC is only able to assign significant centrality to the most

well connected node in the larger of the two subgraphs. As P is raised to 5 and 10, gradually more nodes are assigned significant centrality scores, even some in the smaller subgraph of 50 nodes.

For an illustrated example of PCC applied to a graph whose hubs are less prominent, refer to the example of the geometric mesh graph with a non-scale free distribution in Figure 4 and its graph's degree distribution in Figure 3. A more detailed demonstration of PCC on a graph with a degree distribution not resembling a scale free distribution is available in [16].

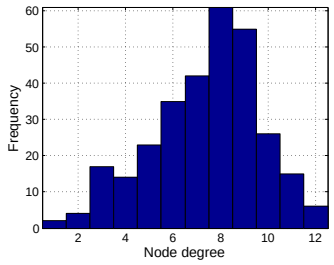


Fig. 3: Degree distribution of a geometric mesh graph of 300 nodes.

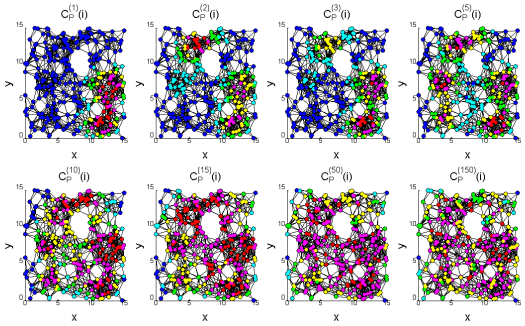


Fig. 4: PCC applied to a geometric mesh graph of 300 nodes. Warm node colors represent high PCC, cold node colors represent low PCC.

E. Selection of Number of Eigenvectors

The cost of computing an eigenvector can be significant for large matrices, favoring the use of as few eigenvectors for PCC as are necessary. To determine appropriate number of eigenvectors (P_{app}), we consider the phase angle ϕ as a function of P. The phase angle $\phi(P)$ of a PCC vector $\mathbf{C}_P$ is defined as its angle with the EVC vector $\mathbf{C}_E$ and is defined mathematically in equation 6.

$$\phi(P) = \arccos\left(\frac{\mathbf{C}_P}{|\mathbf{C}_P|} \cdot \frac{\mathbf{C}_E}{|\mathbf{C}_E|}\right) \tag{6}$$

When the phase angle function is plotted for a range of P, the value of P at which ϕ begins approaching its final steady value is used for that particular graph (P_{app}) [17]. The selection of P_{app} can be made as,

$$P_{app} = min\{\phi(P+1) - \phi(P)\} \quad \in [-\epsilon, \epsilon], \forall\ [P, N], \tag{7}$$

where ϵ is a small real number. It is our observation that the value of P_{app} is close to the number of well-connected communities in a social graph.

F. Decentralized Eigendecomposition Algorithm

The massive sizes of social networks require a method whose space and time complexity scales well with the number of nodes and links between them. According to Equation 3, for a node to compute its own PCC score it needs to know its corresponding entries in the first P eigenvectors $\mathbf{x}_1$, $\mathbf{x}_2$ ... $\mathbf{x}_P$ of the adjacency matrix $\mathbf{A}$, as well as who its neighbors are, i.e. the entries in its corresponding row of A. Although many decentralized algorithms for computing eigenvectors of a matrix exist, many of them are not designed to minimize the communication overhead between participating nodes. We discuss 2 well-known distributed algorithms in the following text.

In [18] Kempe and McSherry developed a decentralized algorithm for the computation of the first P most significant eigenvectors. Their approach differs from other algorithms in that each node is only required to communicate with neighbor nodes. This means that the computational complexity of the algorithm at every node, and the volume of messages exchanged by each node scales only linearly with the number of its neighbors and linearly with the number of eigenvectors that are computed. Furthermore, the time for the algorithm to converge is $O(\tau_{mix} \log N)$, where N is the total number of nodes in the graph and τ_{mix} is the mixing time of the Markov chain with a transition matrix that is the row-normalized version of $\mathbf{A}$. Although the Power method with deflation's ([23], [25]) overhead and convergence properties vary greatly from those of the KM algorithm, the iterative components of the KM algorithm are very similar to those of the power method when it is used in the computation of the principal eigenvector only. Both algorithms perform a deterministic simulation of a random walk. For detailed coverage of the KM algorithm we refer the reader to [18].

The principal advantage of the KM algorithm is its lower message exchange overhead. Given the degree of each node in a graph, the number of messages exchanged network-wide in each iteration of the KM algorithm is deterministic. In each iteration, every node sends a message to all nodes it is connected with. Therefore, the number of messages exchanged is twice the number of edges in the graph.

Kempe reported the error of the ℓ_2 norm of the space spanned by R_P, the projection of P most significant eigenvectors on $\mathbf{A}$, and $R_{P'}$, the projection of P most significant eigenvectors by KM-algorithm onto $\mathbf{A}$, with high probability as follows.

$$\|\mathbf{R}_P - \mathbf{R}_{P'}\|_2 \leq O\left(\left|\frac{\lambda_{P+1}}{\lambda_P}\right|^t \cdot N\right) + 3\epsilon^{4t} \tag{8}$$

Here, t denotes the number of iterations for which the KM algorithm executes. Clearly, since $\lambda_{P+1} < \lambda_P$, the fractional term will be decreasing with t at a geometric rate. Figure 5 shows a plot of average mean squared error (MSE) between the actual and estimated top-k eigenvectors (using the KM

algorithm) for random graphs of 100 nodes. We report average MSE values for varying number of eigenvectors (k) and number of iterations (t). Each point in the plot is an average of 1000 independent runs and the confidence intervals are too small to be shown. As expected from equation 8, we observe that average MSE values sharply decrease approximately at a geometric rate for increasing number of iterations. Furthermore, for a given number of iterations, average MSE values increase for larger k values.

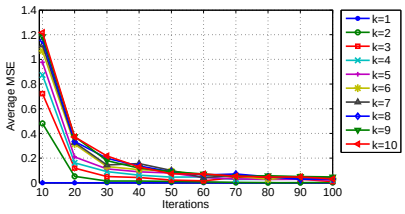


Fig. 5: Average Mean Squared Error (MSE) for the KM algorithm reported for varying values of number of eigenvectors (k) and number of iterations (t).

From the above discussions of the KM algorithm, we conclude the following:

- The KM algorithm never requires a node to communicate beyond its immediate neighbors. This implies that the communication overhead of the KM algorithm scales linearly with the number of computed eigenvectors.
- Kempe *et al.* reported near perfect convergence for their algorithm, which is also verified from our observations in Figure 5.

For these reasons, we choose to use the KM algorithm for the distributed computation of eigenvectors for PCC.

IV. Performance Evaluation

A. Data Sets

We now present details of the data sets used to evaluate the efficacy of our proposed technique.

1) Facebook A and Facebook B: In our study, we use two independently collected data sets from Facebook [34]. We use both data sets to demonstrate that our proposed solution is not biased in favor of any particular data set. The data sets are labeled data set A and data set B here-onwards. As Wilson *et al.* describe in [34], at the time of collection in April 2008 Facebook had 67 million subscribers of whom 44.3 million belonged to a regional network (regional networks were defined on the basis of geography and institutions). Each regional network forms a community of nodes that are strongly intra-connected but sparsely connected to other communities. Their crawler performed a breadth-first-search and collected data from the 22 largest regional networks. The crawler was initialized with 50 randomly seeded user profiles. Wilson *et al.* verified the completeness of their coverage of regional networks by performing 5 simultaneous crawls of the San Francisco regional network, each seeded by a different number of seed user IDs varying from 50 to 5000. The difference in the number of users discovered between crawls was a mere 0.1%. Therefore, we can conclude that the coverage of users in these data sets is fairly complete. The data contained in data set A and data set B is from different regions.

Each data set further consists of two types of graphs. First, we have an undirected friendship graph where the nodes represent users and links represent the friendship between two users. Second, we have a directed pair-wise user interaction graph where the nodes represent users and the directed links represent the interaction from one user to another. The interaction data spans a time duration of one year. Note that we use the interaction data only to evaluate the ground truth.

2) Twitter: The third data set comprises of the follower graph between more than 2 million Twitter users, and the number of different tweets they sent out over the course of the data collection period. This data set, too, consists of two types of data. First, we have an directed follower graph where the nodes represent users and each link represents subscription by a user to another user's Twitter feed. It is different from the two Facebook data sets A & B because the nature of relationships between Twitter users is fundamentally different. Twitter does not have a friendship graph like Facebook; Instead, it has a follower graph of directional, non-reciprocal relationships. Second, we have users' Twitter activity history. It includes all of the following:

Tweets - T: The number of original tweets authored by a user during the data collection period.

Retweets - RT: The number of retweets made by a user during the data collection period.

Retweeted - RTT: The number of times other users retweeted a tweet sourced by the user under consideration.

Tweets+Retweets - TRT: The number of tweets and retweets (T+RT) sent by a user during the data collection period.

The data collection period spans one week. Note that we use the activity history data only as the ground truth.

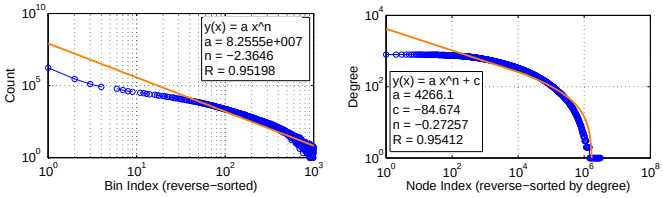
Table I provides the basic statistics of the friendship and follower graphs analyzed in this study. We note that the number of users in data set A are slightly more than those in data set B. We also note that the ratio of the number of friendship links to the number of users for data set A is ~ 7.6, which is slightly more than ~ 7.1 for data set B. The same ratio is an order of magnitude larger for the Twitter data set C. This difference in ratio is reflected in the values of average clustering coefficients of data sets A and B. Moreover, the number of cliques in the friendship graph of data set A is more than those in data set B. However, we observe that the transitivity value (defined as the fraction of possible triangles that are actually triangles) for data set A is less than the respective value of data set B.

Figures 6, 7 and 8 show the plots of degree distributions for graphs of all three data sets. In Figures 6a, 7a and 8a we plot the histograms of one thousand bins in which we have grouped users. Although the distribution does not follow a power-law exactly, it fits it reasonably well as shown by straightness on log-log scale and verified by high goodness-of-fit (R^2) values for data set A and data set B. This observation is in accordance with the result of recent studies that have shown that the degree distribution of many online social networks is power-law [28]. An equivalent representation is shown in Figures 6b, 7b and

TABLE I: Basic statistics of the friendship graphs analyzed in this study

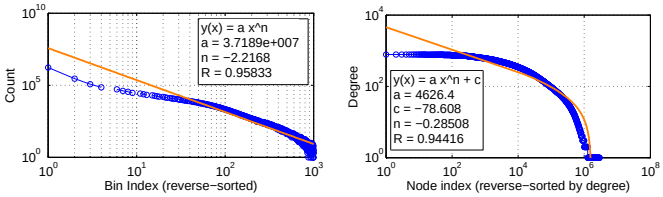
Property	FB Data Set A	FB Data Set B	Twitter Data Set C
# Users	3,097,165	2,937,612	2,082,187
# Friendship Links	23,667,394	20,959,854	102,143,769
Average Clustering Coefficient	0.0979	0.0901	-
# Cliques	28,889,110	27,593,398	-
Transitivity	0.0477	0.04832	-

8b where users are reverse-sorted by their degree. Note that the estimated values of model parameters are similar for data sets A and B.



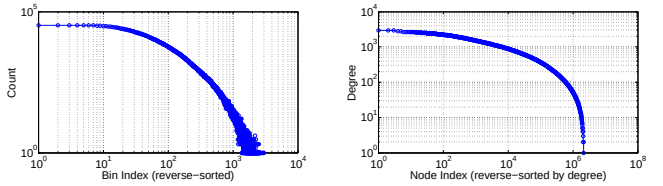
(a) Power law (b) Zipf-like

Fig. 6: Degree distribution of friendship graph for Facebook data set A.



(a) Power law (b) Zipf-like

Fig. 7: Degree distribution of friendship graph for Facebook data set B.

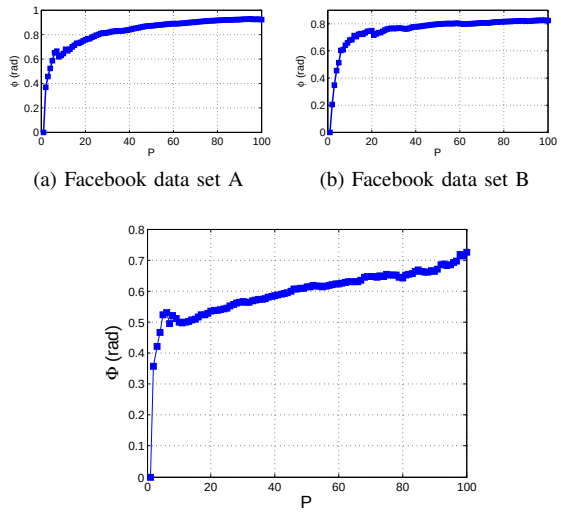


(a) Power law (b) Zipf-like

Fig. 8: Degree distribution of friendship graph for Twitter data set C.

B. Selection of PCC Parameter

We can compute the PCC vector $\mathbf{C}_P$ for a range of number of eigenvectors P. Note that at $P = 1$ the PCC $\mathbf{C}_1$ is the EVC $\mathbf{C}_E$, which serves as the measure of baseline comparison, as mentioned in [32] and [27]. Although we will be comparing PCC with EVC for a range of values of P in some of our subsequent analysis, we will try to determine the "appropriate" number of eigenvectors for PCC (denoted by P_{app}). We do this by means of plotting the phase angle function defined in Equation 6. Figures 9a, 9b and 9c plot the phase angle functions of all three data sets A, B and C, respectively, for the range of $P = 1$ to 100. Incidentally, For data sets A, B and C, the phase angle function rises quickly initially until $P = 6$ and rises only very slowly thereafter. Using Equation 7, the P_{app} values are 10, 20 and 26 for data sets A, B and C, respectively. The difference in P_{app} values for data sets A and B can be explained by differences in their network structures. In terms of PCC computation, this indicates that all nodes can be reached from a given node in lesser number of hops on average. In other words, fewer number of eigenvectors (denoted by P_{app}) are enough to approximate the steady-state PCC value.



(a) Facebook data set A (b) Facebook data set B

(c) Twitter data set C

Fig. 9: Plot of the phase angle $\phi(P)$ between PCC vectors $\mathbf{C}_P$ and EVC vector $\mathbf{C}_E$ plotted against number of feature vectors P for (a) Facebook data set A, (b) Facebook data set B, (c) Twitter data set C.

C. Comparison With Ground Truth

Now that we have identified an appropriate number of eigenvectors for PCC for both data sets, we devote the remaining section to evaluating its accuracy by comparing the results to the ground truth, *i.e.* interaction data. For both data sets A and B, we have interaction graphs spanning 1 month, 6 months, and 1 year time periods. For the Twitter data set C, we have four different activity measurements, namely each user's tweets, retweets, the number of times he/she was retweeted, and the sum of the number of tweets and retweets.

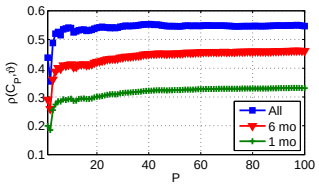
1) Verification of Optimal PCC Parameter: Recall that the PCC scores for individual users are calculated using only information from the friendship graph. We compare the PCC of nodes against their actual flows over various time periods to get a sense of the time period over which PCC best predicts the flow.

We have used a symmetric measure called Pearson's product-moment coefficient to quantify the similarity between the output of PCC and the ground truth from interaction data. The Pearson's product-moment coefficient ρ is defined in Equation 9. Here E is the expectation operator, σ refers to

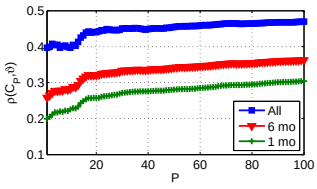
standard-deviation, and μ denotes mean value.

$$\rho(\mathbf{C}_P, \vartheta) = \frac{E[(\mathbf{C}_P - \mu_{\mathbf{C}_P})(\vartheta - \mu_\vartheta)]}{\sigma_{\mathbf{C}_P}\sigma_\vartheta} \quad (9)$$

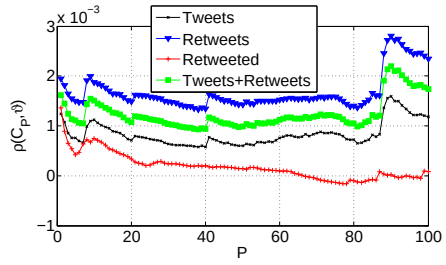
Figure 10 shows the plots of correlation coefficients $\rho(\mathbf{C}_P, \vartheta)$ as a function of number of eigenvectors for the range $1 \leq P \leq 100$. Figure 10a plots $\rho(\mathbf{C}_P, \vartheta)$ for flows collected over 1 month, 6 months and the entire collection time period (labeled 'All') for data set A. Figure 10b does the same for data set B. Figure 10c plots the Pearson's product-moment coefficient of various PCC vectors C_1 through C_{100} with T, RT, RTT and TRT.



(a) Facebook data set A



(b) Facebook data set B



(c) Twitter data set C

Fig. 10: Correlation coefficients ρ of PCC $\mathbf{C}_P$ and, (a) flow count of Facebook data set A ($\vartheta(A)$), (b) flow count of Facebook data set B ($\vartheta(B)$), (c) tweets / retweets / retweeted / tweet+retweet counts of Twitter data set C. The correlation coefficients are plotted as functions of the number of eigenvectors P and plotted separately for each interaction graph.

For the Facebook data sets, we make two major observations from these plots. First, we note that the value of ρ generally increases with increasing number of eigenvectors P for computing PCC. It rises quickly to reach its steady-state value for both data sets. For Facebook data set A, ρ reaches close to its steady-state value at around 10 eigenvectors. Whereas, for Facebook data set B, ρ reaches close to its steady-state value at around 20 eigenvectors. Note that the steady-state values for ρ are reached at P_{app} values selected in the previous subsection. This observation verifies the merit of using phase angle for selection of appropriate value of P in PCC computation.

Second, we note that the correlation coefficients are higher for interaction data collected over longer periods of time. This observation follows our intuition that the trends in short-term interaction data can deviate from our expectations in steady-state friendship graph; However, the trends in long-term interaction data show greater similarity with the underlying friendship graph. This observation remains consistent across both Facebook data sets.

For the Twitter data set the values of ρ change much more erratically with P. More importantly though, we note that RT (the number of times a user retweets) is the user activity that is most correlated with his/her PCC score.

2) Accuracy of PCC in Predicting Top-2000 Users: To further evaluate the accuracy of PCC in finding information hubs, we analyze the overlap between the set of top-2000 users by PCC (denoted by $S_{2000}(\mathbf{C}_P)$) and the ground truth. Note that the choice of 2000 nodes in the following analysis is purely arbitrary. The results of our analysis for different set sizes are qualitatively similar. Let the cardinality of the intersection set of the first k nodes by PCC and the first k nodes by flow/activity ϑ be denoted by $I_k(\mathbf{C}_P, \vartheta)$ and defined in Equation 10 below.

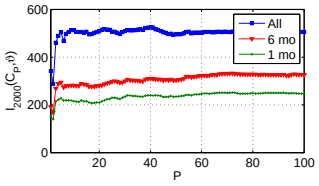
$$I_k(\mathbf{C}_P, \vartheta) = |S_k(\mathbf{C}_P) \cap S_k(\vartheta)| \quad (10)$$

Figures 11a and 11b plot I_{2000} for data sets A and B, respectively. We evaluate separately for interaction data of different durations. As expected, the cardinality of the intersection set increases with the number of eigenvectors used in computation of PCC. In both figures, the data points at $P = 1$ represent the baseline for our comparison, *i.e.* EVC.

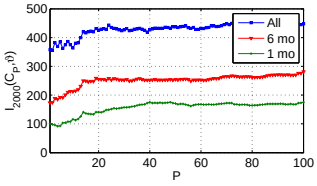
Figure 11c plots I_{2000} for data sets C. Clearly, the cardinality of the intersection sets is an order of magnitude lower than we observe for the Facebook data sets A and B.

For data set A, the cardinality of the intersection set of the top-2000 nodes by EVC and top-2000 nodes by flow ϑ, the cardinality of the intersection set $I_{2000}(\mathbf{C}_E, \vartheta)$ is 342. At $P = 10$, $I_{2000}(\mathbf{C}_{10}, \vartheta) = 513$ for data set A. These numbers represent an increases of 50.0%. For Facebook data set B intersection cardinality of EVC set with flow are $I_{2000}(\mathbf{C}_E, \vartheta) = 358$. At $P = 20$, $I_{2000}(\mathbf{C}_{20}, \vartheta) = 426$ for data set A, an increase of 19.0%. For the remainder of this section, we fix the values of P at P_{app} for both data sets A and B. We see greater agreement between the list of nodes generated by PCC score with flow data collected over a longer durations.

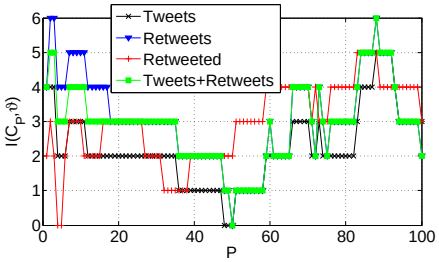
For data set C, the cardinality of the intersection set of the top-2000 nodes by PCC and top-2000 nodes by user activity ϑ is denoted by $I_{2000}(\mathbf{C}_P, \vartheta)$. At $P = 26$, $I_{2000}(\mathbf{C}_{10}, \vartheta)$ for data set A.



(a) Facebook data set A



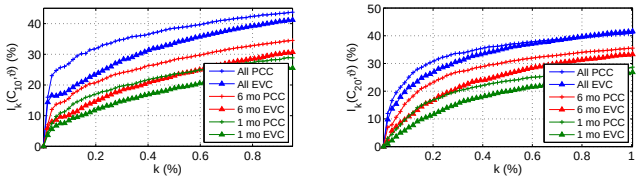
(b) Facebook data set B



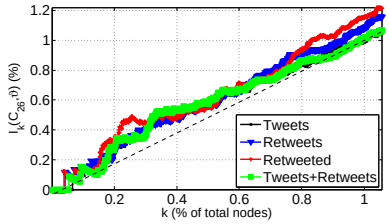
(c) Facebook data set C

Fig. 11: Size of the intersection set in (a) Facebook data set A, (b) Facebook data set B, (c) Twitter data set C, for varying number of eigenvectors used in computation of PCC.

3) Accuracy of PCC in Predicting Top-k Users: Figures 12a, 12b and 12c plot I_k set for top-k users for data sets A, B and C, respectively. We observe an increasing trend for I_k as we increase the bracket size of top-k users. We also note that the cardinality of the intersection set increases for increasing durations of interaction data. The overlap approaches 40% of k for top-1% users. Moreover, we observe that the results for Facebook data set A are slightly better than those of Facebook data set B. The same results for the Twitter data set C, however, are less positive.



(a) Facebook data set A (b) Facebook data set B



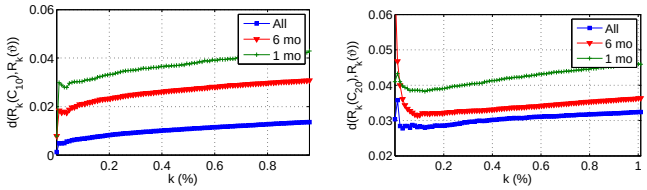
(c) Twitter data set C

Fig. 12: Cardinality of the intersection set in (a) Facebook data set A, (b) Facebook data set B, (c) Twitter data set C, for varying fraction of nodes in graph.

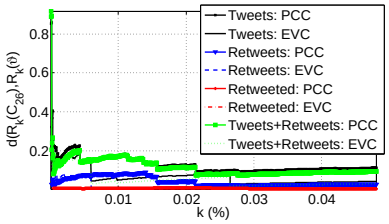
4) Accuracy of PCC in Rank Prediction: The evaluation described till now focuses on the number of users that are common in top-k set assembled with respect to PCC scores and node degree in directed interaction graph. In a more fine-grained analysis, we are also interested in quantifying the accuracy of ranks assigned using PCC scores. Towards this end, we compute the difference between ranks assigned by PCC and those determined using data from interaction graph. Moreover, the significance of correct ranking of high-ranked users is more important than low-ranked users. To accomplish these objectives, we have devised a distance metric to compare the relevance of two ordered lists. We denote the list of nodes of length k in descending order of $\mathbf{C}_P$ by $\mathcal{R}_k(\mathbf{C}_P)$ and the list of nodes of length k in descending order of interaction graph degree by $\mathcal{R}_k(\vartheta)$. The distance is normalized in the range $[0, 1]$, where 0 correspond to the perfect match between two given order lists, and vice-versa. We define the normalized distance $d \in [0, 1]$ between these two ordered lists as:

$$d\left(\mathcal{R}_k(\mathbf{C}_P), \mathcal{R}_k(\vartheta)\right) = \frac{\sum_{i \in \mathcal{R}_k(\vartheta)} \left[\frac{w_i |\mathcal{R}_k(\mathbf{C}_P(i)) - \mathcal{R}_k(\vartheta(i))|}{N - 2i + 1}\right]}{\sum_{i \in \mathcal{R}_k(\vartheta)} w_i} \tag{11}$$

Here w_i is the degree of user i in the interaction graph and N is the total number of users. Figures 13a, 13b and 13c show the variation in distance between two ordered lists as we increase its size k for data sets A, B and C, respectively. Similar to the intersection results for the Facebook data sets, we first note that the best results are achieved when comparison is done with interaction data of longer time duration. Second, we note that the results slightly degrade for increasing values of k. Third, it is evident that the results for Facebook data set A are better than those for Facebook data set B. For example, $d \approx 0.01$ at $k = 0.5\%$ of N for data set A, whereas $d \approx 0.03$ at $k = 0.5\%$ of N for data set B.



(a) Facebook data set A (b) Facebook data set B



(c) Facebook data set C

Fig. 13: Distance between ordered lists computed by PCC and interaction data using (a) Facebook data set A, (b) Facebook data set B, (c) Twitter data set C, for varying fraction of nodes in graph.

V. Conclusions

Information hubs in social networks play important roles in the speed and depth of information diffusion. Identifying hubs helps us to harness their power to pursue social good at local, national, and global levels. In this paper, we propose the first friendship graph based, fully distributed, and privacy-preserving method for identifying hubs in online social networks. Unlike prior work, our method can be used to identify hubs by parties other than social network owners as we do not require any entity to access interaction or friendship graphs. We evaluated PCC and Generalized PCC using data collected from Facebook and Twitter. The Facebook data sets used in this study were collected over the period of more than a year and contain data from about 6 million users. The Twitter dataset was collected over a shorter period of time of only 1 week. The results of our analysis on the Facebook data sets showed that our proposed approach better (in terms of number of correctly identified top-k nodes and their estimated rank) identifies the top-k information hubs in a social network. For the Twitter dataset we used the generalized form of the PCC. The results on the Twitter dataset still demonstrate an improvement vs. random selection of top-k nodes as well as EVC, but only slightly. Based on the trend in the Facebook data set where results correlated better with user behavior over longer periods of time, we explain the difference in performance by the fact that that the Twitter data set was collected over a very short period of one week.

References

[1] *Facebook Advertising, http://www.facebook.com/advertising/*, 2010.

[2] *Facebook Pages, http://www.facebook.com/advertising/*, 2010.

[3] "Statement of rights and responsibilities," *http://www.facebook.com/terms.php*, 2010.

[4] A. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, p. 509, 1999.

[5] B. Bollobas, *Modern graph theory*. Springer Verlag, 1998.

[6] P. Bonacich, "Factoring and weighting approaches to status scores and clique identification," *Journal of Mathematical Sociology*, vol. 2, no. 1, pp. 113–120, 1972.

[7] ——, "Technique for analyzing overlapping memberships," *Sociological Methodology*, vol. 4, pp. 176–185, 1972.

[8] S. Borgatti, "Centrality and network flow," *Social Networks*, vol. 27, no. 1, pp. 55–71, 2005.

[9] G. Caldarelli, *Scale-Free Networks*. Oxford University Press, 2007.

[10] G. Canright, K. Engø-Monsen, and M. Jelasity, "Efficient and robust fully distributed power method with an application to link analysis," *Department of Computer Science, University of Bologna, Tech. Rep. UBLCS-2005-17*, pp. 2005–17, 2005.

[11] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, 2nd ed. Wiley New York, 2001.

[12] P. A. Estevez, P. Vera, and K. Saito, "Selecting the most influential nodes in social networks," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, 2007.

[13] D. Geere, "Samsung offers free phones to frustrated iPhone users," *CNN Tech*, vol. http://www.cnn.com/2010/TECH/mobile/07/24/ samsung.replacing.iphones/, 2010.

[14] J. Goldenberg, S. Han, D. R. Lehmann, and J. W. Hong, "The role of hubs in the adoption processes," *Journal of Marketing, American Marketing Association*, 2008.

[15] A. Goyal, F. Bonchi, and L. Lakshmanan, "Discovering leaders from community actions," in *Proceeding of the 17th ACM Conference on Information and Knowledge Management (CIKM)*, 2008.

[16] M. Ilyas and H. Radha, "A klt-inspired node centrality for identifying influential neighborhoods in graphs," in *Information Sciences and Systems (CISS), 2010 44th Annual Conference on*. IEEE, 2010, pp. 1–7.

[17] M. U. Ilyas and H. Radha, "A KLT-inspired node centrality for identifying influential neighborhoods in graphs," in *Proceedings of the 44th Annual Conference on Information Sciences and Systems (CISS)*, 2010.

[18] D. Kempe and F. McSherry, "A decentralized algorithm for spectral analysis," *Journal of Computer and System Sciences*, vol. 74, no. 1, pp. 70–83, 2008.

[19] D. Kempe, J. Kleinberg, and E. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data mining (KDD)*, 2003.

[20] M. Kimura, K. Saito, and R. Nakano, "Extracting influential nodes for information diffusion on a social network," in *Proceedings of the 22nd National Conference on Artificial Intelligence (AAAI)*, 2007.

[21] M. Kimura, K. Saito, R. Nakano, and H. Motoda, "Finding influential nodes in a social network from information diffusion data," *Springer Social Computing and Behavioral Modeling*, 2009.

[22] C. Kohlschütter, P. Chirita, and W. Nejdl, "Efficient parallel computation of pagerank," *Proceedings of the 28th European Conference on IR Research (ECIR)*, pp. 241–252, 2006.

[23] C. Lanczos, "An iteration method for the solution of the eigenvalue problem of linear differential and integral operators," *Institute for numerical analysis*, 1949.

[24] A. Langville, C. Meyer, and P. FernÁndez, "Googles pagerank and beyond: the science of search engine rankings," *The Mathematical Intelligencer*, vol. 30, no. 1, pp. 68–69, 2008.

[25] R. Lehoucq and D. Sorensen, "Deflation Techniques for an Implicitly Restarted Arnoldi Iteration," *SIAM Journal on Matrix Analysis and Applications*, vol. 17, p. 789, 1996.

[26] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance, "Cost-effective outbreak detection in networks," in *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data mining (KDD)*, 2007.

[27] P. Marsden, "Egocentric and sociocentric measures of network centrality," *Social Networks*, vol. 24, no. 4, pp. 407–422, 2002.

[28] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, "Measurement and analysis of online social networks," in *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement (IMC)*, San Diego, CA, October 2007.

[29] B. Ruhnau, "Eigenvector-centrality–a node-centrality?" *Social networks*, vol. 22, no. 4, pp. 357–365, 2000.

[30] K. Sankaralingam, S. Sethumadhavan, and J. Browne, "Distributed pagerank for p2p systems," in *Proceedings of the 12th IEEE International Symposium on High Performance Distributed Computing (HPDC)*, 2003.

[31] R. Shahbaz, "how high is your popularity?" *http://www.facebook.com/apps/application.php?id=174042725891*, 2010, 12120 monthly users.

[32] X. Shi, M. Bonner, L. Adamic, and A. Gilbert, "The very small world of the well-connected," in *Proceedings of the 19th ACM conference on Hypertext and hypermedia (HT)*, 2008.

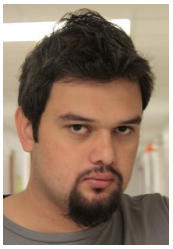
[33] N. R. Suri and Y. Narahari, "Determining the top-k nodes in social networks using the Shapley value," in *Proceedings of the 7th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2008.

[34] C. Wilson, B. Boe, A. Sala, K. Puttaswamy, and B. Zhao, "User interactions in social networks and their implications," in *Proceedings of the ACM European Conference on Computer Systems (EuroSys)*, 2009.

[35] E. Zheleva and L. Getoor, "To join or not to join: The illusion of privacy in social networks with mixed public and private user profiles," in *Proceedings of the 18th International World Wide Web conference (WWW)*, 2009.

[36] F. Zou, Z. Zhang, and W. Wu, "Latency-bounded minimum influential node selection in social networks," in *Proceedings of the 4th International Conference on Wireless Algorithms, Systems, and Applications (WASA)*, 2009.

Muhammad U. Ilyas did his Post-doctoral work at the Department of Computer Science & Engineering and later jointly with the Department of Electrical & Computer Engineering at Michigan State University (MSU) from 2009-2011. He received his Ph.D. and MS Electrical Engineering from MSU in 2009 and 2007, respectively. He received his MS Computer Engineering from the Lahore University of Management Sciences (LUMS), Lahore, Pakistan in 2005, and his BE Electrical Engineering from the National University of Sciences & Technology (NUST) in 1999. He is currently an Assistant Professor in the Department of Electrical Engineering at the School of Electrical Engineering & Computer Science (SEECS) of the National University of Sciences & Technology. His research interests include system modeling and measurement, social network analysis, networking, algorithms, and security.



M. Zubair Shafiq received his B.E. degree in Electrical Engineering from the National University of Sciences and Technology (NUST), Pakistan, in 2008. He is currently pursuing the Ph.D. degree in computer science at Michigan State University. He was with Next Generation Intelligent Networks Research Center (nexGIN RC), Pakistan as a researcher from 2007 to 2009. His research interests include measurement and modeling of cellular networks and online social networks, and computer and network security.

Alex X. Liu received his Ph.D. degree in computer science from the University of Texas at Austin in 2006. He is currently an assistant professor in the Department of Computer Science and Engineering at Michigan State University. He received the IEEE & IFIP William C. Carter Award in 2004 and an NSF CAREER award in 2009. He received the MSU College of Engineering Withrow Distinguished Scholar Award in 2011. His research interests focus on networking, security, and dependable systems.

Hayder Radha received the Ph.M. and Ph.D. degrees from Columbia University (1991 and 1993). He is a Professor of Electrical and Computer Engineering (ECE) at Michigan State University (MSU). He was a Philips Research Fellow and a Distinguished Member of Technical Staff at Bell Laboratories. Dr. Radha is an IEEE Fellow. He is an elected member of the IEEE Technical Committee on Image, Video, and Multidimensional Signal Processing (IVMSP) and the IEEE Technical Committee on Multimedia Signal Processing (MMSP).