Your Best might not be Good enough: Ranking in Collaborative Social Search Engines

Prantik Bhattacharyya*, Jeff Rowe*, Shyhtsun Felix Wu*‡, Karen Haigh†, Niklas Lavesson‡ and Henric Johnson‡
*Department of Computer Science, University of California, Davis, Email: {pbhattacharyya,jbrowe,sfwu}@ucdavis.edu
†Intelligent Distributed Computing Group, BBN Technologies, Email: khaigh@bbn.com
‡School of Computing, Blekinge Institute of Technology, Email: {Niklas.Lavesson,Henric.Johnson}@bth.se

***Abstract*—A relevant feature of online social networks like Facebook is the scope for users to share external information from the web with their friends by sharing an URL. The phenomenon of sharing has bridged the web graph with the social network graph and the shared knowledge in ego networks has become a source for relevant information for an individual user, leading to the emergence of social search as a powerful tool for information retrieval. Consideration of the social context has become an essential factor in the process of ranking results in response to queries in social search engines. In this work, we present *InfoSearch*, a social search engine built over the Facebook platform, which lets users search for information based on what their friends have shared. We identify and implement three distinct ranking factors based on the number of mutual friends, social group membership, and time stamp of shared documents to rank results for user searches. We perform user studies based on the Facebook feeds of two authors to understand the impact of each ranking factor on the result for two queries.**

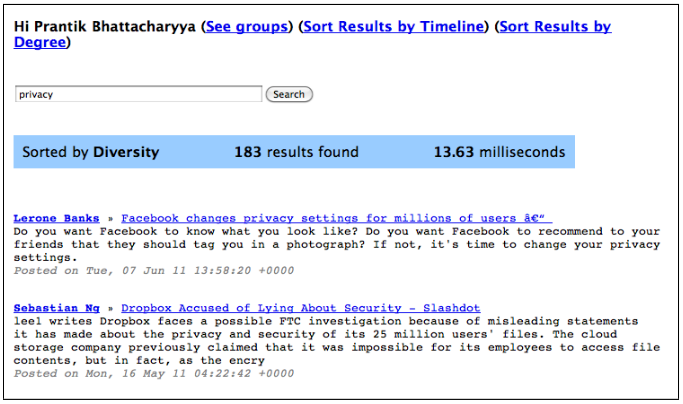
KEYWORDS

Social Networks, Search Engine, Social Search, Ranking

I. INTRODUCTION

Users in online social networks have surpassed hundreds of millions in numbers. With this staggering growth in the network size, social network platform providers like Facebook and Twitter have introduced new tools to engage the users. In addition to connecting and exchanging messages with friends on a regular basis, users can also share photos, videos and location information. Users of online social network platforms also have the unique ability to share with their friends URLs to websites and web articles they read, enjoy and find useful. Social network platforms represent a place to share and reflect what users value greatly and find useful, so people are becoming very good at sharing exactly what they value as well as the pieces of information that are beneficial to them and their friends can also gain from. Facebook has introduced 'Like', 'Share' and 'Recommend' buttons that content providers of any website can include on their website to help visitors share the URLs with their friends in a fast and easy manner. Twitter has also introduced similar technologies to let users 'Tweet' the URL in addition to their personal comment about the URL. The simplicity and ubiquitousness of this technology has propelled the integration of the web graph with the social graph.

Users share their favorite webpages on current affairs, news, technology updates, programming, cooking and so on with their friends using the social network platform. By utilizing the knowledge present in each individual's personal network, a social context based search engine can significantly impact the way information is searched and retrieved. In the current state of web search engines, users are restricted to search for information from the global web and retrieve results that search engine algorithms rank as relevant. The search engines retrieve and rank information based on their understanding of the link structure that relate one webpage to another and assigning a random piece of information a certain value for the purpose of ranking. The only form of authenticity a user can access during the retrieval process is the ranking value assigned by the search engine except for which the retrieved information is only a random piece of information from the web. However, the result set can be significantly changed by introducing social context and social authenticity as factors during the ranking procedure to let querying users identify results based on the way friends have shared information in similar contexts. In this regard, the volume of shared information has grown so rapidly that search engine platforms like Google and Microsoft has also started to introduce signals in their ranking algorithms that reflect the patterns of information share across the social graph [1], [2]. The efforts of search engine platforms has primarily concentrated on introducing the signals incorporated from social sharing to add a parameter to



Hi Prantik Bhattacharyya (See groups) (Sort Results by Timeline) (Sort Results by Degree)

Sorted by **Diversity** **183** results found **13.63** milliseconds

Lerone Banks » Facebook changes privacy settings for millions of users â€"
Do you want Facebook to know what you look like? Do you want Facebook to recommend to your friends that they should tag you in a photograph? If not, it's time to change your privacy settings.
Posted on Tue, 07 Jun 11 13:58:20 +0000

Sebastian Ng » Dropbox Accused of Lying About Security - Slashdot
leel writes Dropbox faces a possible FTC investigation because of misleading statements it has made about the privacy and security of its 25 million users' files. The cloud storage company previously claimed that it was impossible for its employees to access file contents, but in fact, as the encry
Posted on Mon, 16 May 11 04:22:42 +0000

Figure 1: Screenshot of InfoSearch Application on Facebook: Results for the query 'privacy' appear for one of the authors.

COLLABORATECOM 2011, October 15-18, Orlando, United States

DOI 10.4108/icst.collaboratecom.2011.247154

the ranking algorithms for search results that are identical for all users with respect to a specific query. However, the growth of information sharing in individual social network of users can also be used to provide exclusive results for queries from each individual user based on the large volume of information that is available in their network. The motivation to provide such exclusive search results is to incorporate context and authenticity of information source as present in the information shared by users of social network platforms in their personal social networks. The search process thus not only enables a user to access a set of information that has a distinct social component attached to it but also to gain from the collective knowledge of their respective social network. In other words, a search process is no longer limited to retrieving a random piece of information from the Internet with no trust value attached to it but extends to a retrieval process that includes a trusted source, that is, their friends' personal attachment or endorsement of that piece of information.

Furthermore, in a social network, user connections can also be seen as a graph such that a user can be represented as a node and each friend connection can be treated as an edge between two nodes. However, link analysis algorithms like PageRank [3], [4], [14] are not suitable for application here since during the search process of an individual user, results from the members of their social circle should not be ranked based on a generalized analysis of the relative importance of those members in the larger network but rather on their local importance to the querying user [16], [17].

In this work, we consider the process of information retrieval from social network of individual users and algorithms for ranking these results as part of the social search problem. We have developed the social search engine on the Facebook platform. The search engine is called *InfoSearch* and is available at http://apps.facebook.com/infosearch. We use three different algorithms to rank search results. First, we use the time property of a shared information to rank results in a chronological manner. Second, we use the number of friends shared between the user performing the query and the user who acts as a source for the shared information. For our third approach, we utilize the social relationship between the user performing the query and the user who shares the information to rank results and form the final result set. We derive the social relationship between two users based on the social group structure shared between them. We present results based on the impact of the above three ranking algorithms in retrieving information during the case studies.

The paper is organized as follows. In Section III, we formally describe the problem statement related to social search and follow up with a discussion of social network relationship semantics. In Section V, we discuss the ranking algorithms employed during our system development and Section VI presents the architecture of the social search engine. In Section VIII, we present our findings obtained through user studies and the last section closes with a discussion on future research directions.

II. Related Work

Several projects have looked into the area of search in social networks. The research problems have broadly fallen into the following categories. First, the identity or profile search problem in which social network information is used to connect and subsequently search for users. Dodds et. al. [5] conducted a global social-search experiment to connect $60,000$ users to 18 target persons in 13 countries and validated the claims of small-world theory. Adamic et. al. [6] conducted a similar project on the email network inside an organization.

In the second category, social networks have been leveraged to search for experts in specific domains and find answer to user questions. Lappas et. al. [7] addressed the problem of searching a set of users suitable to perform a job based on the information available about user abilities and compatibility with other users. The work in [8] attempted at automated FAQ generation based on message routing in a social network through users with knowledge in specific areas. Other works in similar directions have also been presented, e.g. [13], [15]. Query models [9] based on social network of users with different levels of expertize for the purpose of decentralized search have also been developed. Horowitz et. al. [10] presented *Aardvark*, a social network based system to route user questions into their extended network to users most likely knowledgeable in the context of the question.

In the third category, social networks are considered to improve search result relevancy. Haynes et. al. [11] studied the impact of social distance between users to improve search result relevancy in a large social networking website, *LinkedIn*. The author defines the social distance between users based on the tie structure of the social graph and aims to provide improved relevance and order in profile identity entries. Mislove et. al. [12] consider the problem of information search through social network analysis. They compare the mechanisms for locating information through web and social networking platforms and discuss the possibility of integrating web search with social network through a HTTP proxy. In this work, we build the social search system on Facebook, utilizing the existing social graph as well as the knowledge database already being build by its users. We discuss the details next.

III. Social Search - Problem Statement

In this section, we illustrate how users share information and discuss the benefits of information sharing. Next, we introduce the social search problem statement addressed in this work.

By sharing the URL on Facebook, the user is introducing the article to his friends, extending the knowledge database of the social network with the context of the article. In this case, the context of the article is 'privacy' and other related keywords. Users in the network benefit from this shared knowledge when they try to find information related to privacy. Furthermore, the social context in this case i.e. the person who shared this information can help the querying users to disambiguate and choose from the large number of articles available on 'privacy' in general on the web. Thus, users benefit from the fact that someone in their social network already read the article and shared with their friends for the article's relevancy. Next, we formally define the task of information retrieval and ranking in a social search engine. We begin with a few key information structures.

Definition 1. ***Social Network.*** *A social network is a graph $G = (V, E)$, where V is a set of nodes and E is a set of edges among V. A node stands for a user in the social network, and an edge e stands for a connection between two users u and v. In our work, we consider undirected edges. The shortest geodesic distance between two nodes n_1 and n_2 in the network is defined as $d(n_1, n_2)$. Let $d(n_1, n_2) = \infty$ if no path exists between the nodes in the network.*

Definition 2. ***Ego Network.*** *For a user u, ego network is a graph $G(u) = (V(u), E(u))$, where $V(u)$ is a set of nodes that includes all friends of u, $F(u)$ and the node u itself. $E(u)$ is a set of edges among $(V(u)-u)$ such that $\forall v \in (V(u)-u)$, v and u are friends and share an edge in E. Additionally, all edges between nodes in $(V(u) - u)$ that existed in E are also included in $E(u)$.*

Definition 3. ***Mutual Friend Network.*** *A mutual friend network of an user u is defined as a subset of the ego network, represented as $MF(u) = (F(u), E'(u))$. $F(u)$ is the set of all friends of user u and $E'(u)$ is a subset of the edges from $E(u)$ with the edges between user u and nodes in $F(u)$ absent.*

Definition 4. ***Shared Document.*** *An URL or document shared by a user u is identified by the tuple (u, d). Each shared URL or document is tagged by a set of keywords $K(d) = (k_1^d, k_2^d, ..., k_m^d)$. Additionally, each document is also tagged by a timestamp, $T(d)$, based on the time the document was shared by the user in the social network platform.*

Definition 5. ***Result Candidates*** *A query with the keyword q by a user u is defined as $Q(u, q)$. The result candidates for the query $Q(u, q)$ is defined as the set of shared document tuples, $RC(Q(u, q)) = (v_i, d_j)$ such that $v_i \in F(u)$ and $\forall d_j, q \in K(d_j)$.*

Definition 6. ***Results Final.*** *For a query $Q(u, q)$, the final result set $RF(Q(u, q))$, of ρ results, is formed from the $RC(Q(u, q))$ possible result candidates. The final result set of ρ results is determined by the contribution of each v_i from the set of result candidates such that each user v_i who acts as a source of information can impart social context into the result set.*

In the next section, we will discuss and define the semantics of social relationship such that we can formalize the contribution of each user as they impart social context in formulating the final result set. We will introduce the ranking factors and the algorithm to determine the final result set in section V.

IV. Semantics of Social Relationships

One way to understand the relationship between users u and v is to understand the number of connections user v share with other users in the mutual friend network. This number indicates the degree of user v in $MF(u)$ i.e. the factor that indicates how many users in $F(u)$ connect to v. The other way to understand the relationship between two users is to empirically determine the social groups of user u from the mutual friend network and use this information to understand v's relation to other users in $F(u)$, including those users not directly connected to v. In the next subsections, we introduce the formal definitions of each factor.

1) Shared Mutual Friends: In this factor, we consider the degree of user v in $MF(u)$ i.e. the factor that indicates the number of users in $F(u)$ connect to v. Let, this value be represented as $mf(v, MF(u))$, for all $v \in F(u)$. We present an example in Figure 2. Ego e is connected to all the other nodes in the graph and shown using a broken line between the vertices and ego e. The mutual friend network of the ego e is shown by the connected lines between the other vertices of the figure. In this example, the number of shared mutual friends for vertex a with respect to ego e is 2. Similarly, vertices b and c has 2 and 3 shared mutual friends respectively. The number indicates the strength of connectivity of a particular vertex or friend in the mutual friend network for a user and thus is an important signal to represent the social relationship shared between users.

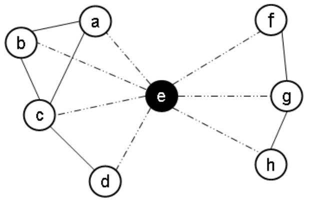


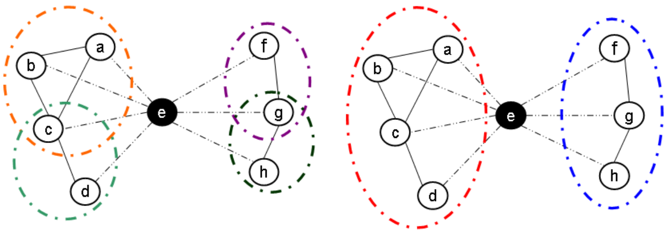
Figure 2: Example ego-network of ego e

2) Social Groups: A social group in the ego-network of user u can be defined as a set of friends who are connected among themselves, share a common identity and represents a dimension in the social life of the user u. A social group can be defined in multiple ways. In this work, we base our definition on mutuality [17] and the formal definition is presented next.

Definition 7. ***Social Group.*** *A social group of a user u is defined as $sg(u) = (V'')$ where V'' is the set of vertices such that $V'' \subseteq F(u)$ and for two users v and w in V'', $d(v, w) \leq k$ in the mutual friend graph, $MF(u)$. The set of all such social groups formed from the mutual friend graph of a user u is represented as $SG(u)$.*

The above definition allows for duplication of users across different social groups since a user can belong to multiple social groups as it satisfies the geodesic requirement with other users of each group.

Let user u's social circle be divided into a set of groups represented as $SG_u = \{sg_u^i\}$, where $1 \leq i \leq ng_u$, ng_u represents the number of social groups formed. Based on two different parameter values, examples of such groups are presented in Figure 3a and Figure 3b. We observe that four social groups are discovered for $k = 1$. Nodes c and g overlap in two groups each. Now, when we inspect the graph for



(a) $k = 1$

(b) $k = 2$

Figure 3: Social Groups for an ego e

$k = 2$, we discover only 2 social groups with no overlapping vertices. It is also important to note here that further increase in the value of k has no effect in group generation. Thus, in a way the group formation gives a sense of separation or distance between the users based on the value of k for the group formation process.

We use the set of all social groups formed from the mutual friend graph of an user u to next define the social distance between two users present in the ego network of user u. Let user v belong to the set of social groups g_v such that $g_v \subset SG(u)$. Let η_v represent the cardinality of g_v and let each element of set of groups g_v be represented as g_v^i such that $1 \leq i \leq \eta_v$. We utilize the group member information to next define group distance and user distance.

Definition 8. ***Social Group Distance.*** *The distance between two social groups is defined to be equal to the Jaccard distance between the groups. For two social groups, $sg(u)_i$ and $sg(u)_j$, from the set $SG(u)$ of user u:*

$$dist(sg(u)_i, sg(u)_j) = 1 - \left(\frac{|sg(u)_i \cup sg(u)_j| - |sg(u)_i \cap sg(u)_j|}{|sg(u)_i \cup sg(u)_j|} \right) \quad (1)$$

Definition 9. ***User Distance in Ego Network.*** *User distance between two users, v and w, in the ego network of user u is defined as the mean distance between the two user's associated group(s). For users v and w associated with η_v and η_w number of social groups represented by g_v^i and g_w^j such that $1 \leq \eta_v$ and $1 \leq \eta_w$ respectively, user distance is defined as:*

$$\omega(v, w) = \frac{\sum_{\substack{1 \leq i \leq \eta_v \\ 1 \leq j \leq \eta_w}} dist(g_v^i, g_w^j)}{\eta_u \times \eta_w} \quad (2)$$

Based on the above two factors to identify social relationship semantics between two users, we next identify the ranking factors and define the ranking algorithms to formulate results in a social search engine.

V. Ranking Factors and Algorithm

In this section, we describe the ranking factors involved to determine $RF(Q(u, q))$ of ρ results from the set of result candidates, $RC(Q(u, q))$. We identify three ranking factors for the purpose. Two factors are based on the social relationship semantics we identified in the previous section and the third factor is based on the time stamp at which the user shared an URL or a document. We also describe the algorithm employed during the ranking process to determine the top results for each factor as we define each factor.

1) Degree: The 'degree' factor is based on the number of mutual friends between two users as described in Section IV.1. For this factor, the algorithm to determine the final result set consists of two steps. First, from the result candidates, $RC(Q(u, q))$, we identify the unique set of users in the possible candidates and then sort them based on the number of mutual friends they share with user u i.e. $mf(v, MF(u))$ from high to low. The first ρ users from the sorted list is selected to construct $RF(Q(u, q))$. If multiple documents associated with the same user exists, we select the document with the most recent $T(d)$ value. If the number of unique users is less than ρ, we then repeat the same steps including other entries by the same users till we reach ρ results.

2) Diversity: The 'diversity' factor is based on the social group information of the querying user. The purpose of this factor is to maximize the group representation in a result set such that the social diversity in a result set is maximized and a higher user distance between the users present in the result set can help user u to inspect results that members from the various groups of the network share on the platform. The diversity value is based on the user-distance methods defined previously and is defined next.

Definition 10. ***Diversity.*** *The diversity of a result set, R, consisting of ρ results is defined as the mean user distance(s) between each pair of users.*

$$\triangle(u, R) = \frac{\sum_{v, w \in R} \omega(v, w)}{|\rho|^2} \quad (3)$$

To determine $RF(Q(u, q))$ based on the diversity factor, the ranking algorithm starts by first constructing a set of unique users from the set $RC(Q(u, q))$. From the set of unique users, the next step consists of forming all possible combinations of ρ users. We denote the set of all combinations as $\mathbb{R}$. In the next step, for all $R_i \in \mathbb{R}$, we determine their social diversity value and present the combination with the highest diversity value as the final result. If the number of unique users in the set of result candidates is less than ρ then the final result set is formed by using the most recently shared document of each user and repeating the process untill ρ results are formed. In this case, a diversity value of 0.0 is associated with the returned result set.

3) Time: With this factor, we consider the time stamp, $T(D)$ at which an article was shared in the network to rank the results. This factor considers the time relevancy of shared documents. For instance, in the context of 'budget', time relevancy leads to information pointed towards current budget issues shared on the social network.

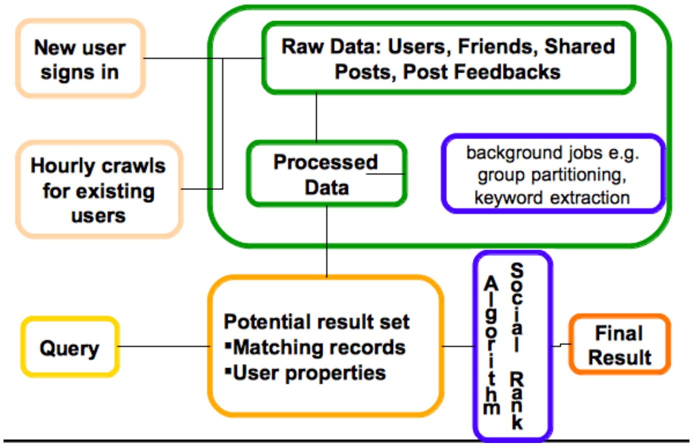

New user signs in
Hourly crawls for existing users
Raw Data: Users, Friends, Shared Posts, Post Feedbacks
Processed Data
background jobs e.g. group partitioning, keyword extraction
Query
Potential result set
▪Matching records
▪User properties
Algorithm
Social Rank
Final Result

Figure 4: Social Search Engine Architecture

The algorithm that ranks results using this factor is straightforward: all documents present in the result candidate set is sorted based on their time time stamp information with the most recent document as the first element. From the sorted list, the first ρ results are considered as the final result set, $RF(Q(u, q))$. In the next section, we discuss our development the social search engine.

VI. Social Search System Development

We built *InfoSearch* as a prototype social search engine over Facebook. *InfoSearch* is built as a Facebook application using the Facebook platform APIs and is available at http://apps.facebook.com/infosearch. Users are requested to authorize the application in order to use it. Once authorized, the three primary components of the application crawl, index and formulate search results. These components are described next. The system architecture for the search engine is presented in Figure 4.

A. Crawler

The purpose of the Crawler is to pull out information from the Facebook feed of each signed-in user using the Facebook API. The Facebook feed of a user consists of links, photos, and other updates from friends. In this work, the Crawler focuses on crawling the shared links to connect the web graph with the social graph. The Crawler is executed on a daily basis for each authorized user to retrieve the aforementioned updates from their feed.

In our work, the Crawler employs the 'links' API provided by Facebook to crawl the various 'links' shared by users over the Facebook platform. When called by the Crawler, the 'links' API returns a set of fields related to each link entry. Among the returned fields, we consider the following fields: a) 'id', b) 'from', c) 'link', d) 'name', e) 'description', f) 'message' and g) 'created_time' for the next component of our search engine. The Crawler also retrieves information about a user's friend list to build the ego and mutual friend network of a user. The Crawler uses the 'friends' and 'friends.getMutualFriends' API to retrieve information about the nodes and edges, respectively to build the ego network of a user. The Crawler also provides the scope to expand our architecture for other social network platforms by mapping the field lists of each returned link with the fields used by the next two components of the application. It is simple enough to extend the application to work with other platforms, since only one change in how the crawler interacts with the platform needs to be introduced.

B. Indexer

The Indexer has two primary tasks. First, it analyzes the information retrieved by the Crawler to build an index of keywords for each shared URL. Second, the Indexer also performs the task of building the social groups for each user. Details of each task are described next.

Once the shared URLs are retrieved from the feed of each signed-in user, the next step is to build a keyword table for each URL with keywords extracted from the text retrieved from the URL. We use Yahoo!'s term extraction engine [18] for this purpose. The term extraction engine takes a string as input and outputs a result set of extracted terms. Additionally, we also use the Python-based topia.termextract library [19] to expand the keyword table. This library is based on text term extraction using the parts-of-speech tagging algorithm. We retrieve text from each URL and interpret the text using the aforementioned methods to finalize the set of keywords for each shared link.

The second task of the Indexer is to analyze the mutual friend network of each signed-in user and build the social group information set. We use the 'R' implementation of the 'kCliques' for this purpose which is focused on the definition for social network analysis [20]. For the user study in Section VIII, we vary the value of k between 1 and 6 to understand the impact of social group formation in the final result formulation. To formulate results for users performing queries through *InfoSearch*, we use a value of k equal to 3.

C. Result Formulation

This is the final component in the system development. The purpose of this component is to a) process the user input queries, b) determine the result candidates, and c) formulate the final result set. In the first step, the user enters a query through the search engine web interface. At this step, users are also given the option to select their preferred way of ranking the possible results. In the next step, all documents related to the input query that originated from the friends of the user are retrieved. If no related documents are found and the query includes multiple keywords, the query is broken into multiple sub-queries and the search process is repeated to determine the related documents. If no documents are found at this stage, a 'no results found' message is sent to the user and the process stops. Otherwise, the set of related documents are promoted to potential result candidates and sent for processing by the ranking algorithms to determine the final result set. Based on the ranking factor selected by the user, the corresponding ranking algorithm is applied to the result candidates and the final result set is pushed forward to the application interface for display to the user.

Table I: User Statistics Collected During 2011

Statistic	Mar 28	Apr 28	May 28
Users crawled	1, 374	2, 134	2, 796
Links Analyzed	12, 464	17, 139	22, 266
Keywords Extracted	487, 706	676, 854	865, 067
Unique Keywords	76, 158	97, 704	115, 570

In our current implementation, we display a set of 8 results as the top results, that is, we consider a value of ρ equal to 8 during the final result formulation step. We implement a pagination style such that after every ρ results are displayed, the next set of top ρ results are determined from the remaining result candidates and the process is repeated until the number of results in the candidate set is less than ρ. Thus, the result sets are displayed to the user in the form of consecutive pages.

We also implement an additional feature to help users find information related to a specific friend or set of friends. This feature is implemented at the query interpretation step and the user has to specify the name of his/her friends in conjunction with the query. In this particular situation, the retrieval process is limited to the set of information related to the specified user(s) only and the $time$ factor is used to rank the results at this step.

VII. User Statistics

We invited colleagues from our laboratory to use the application. This step has helped us crawl their Facebook feeds to collect data in order to understand the impact of each ranking factor on the result formulation. In this section, we present a few statistics on the collected data. Currently, *InfoSearch* has 15 signed-in users and through their Facebook feed, it has access to regular updates of $3,650$ users. Each user has an average of 243 users in their ego network and their mutual friend graph has an average of 1655 edges. We present statistics on the number of links shared by members of the ego network in Table I.

During the time *InfoSearch* has been active, we have crawled links shared by $2,796$ users. This is a very significant number because it tells us that, among the users *InfoSearch* has access to, 76.70% shared a web link with their friends in the social network. It is evident that the integration of web and social network graphs is taking place at a rapid pace and that the growth can have a significant impact on the way users search for information on the Internet.

The number of links shared by the users during this period is $22,266$ and growing. The number of keywords extracted using the Yahoo! term extraction engine and the Python topia.termextract library is $865,067$, which amounts to an average of 39 terms for each link. Additionally, we also consider the number of unique terms present in this pool to form a picture about the uniqueness in the shared content. We observe that the number of unique terms shared across all the links is $115,570$, which results in an average of 5 terms per link. We next discuss case studies to understand the performance of social search engine results under different ranking factors and algorithms.

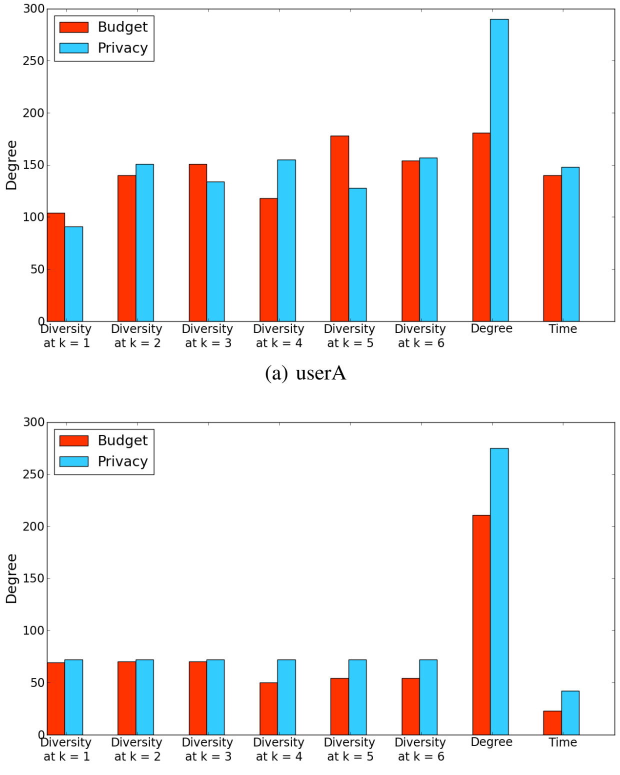
VIII. User Studies

The subjectiveness of a social search engine, that is focused on unique search results for each user based on their respective ego network, makes it intrinsically difficult to compare and contrast the quality of the final results produced with the results of other search engines where similar or identical results are generated for all users. Thus, we cannot evaluate the results shown by *InfoSearch* for a query based on the results obtained from other web search engines. Instead, we focus on analyzing the impact of the ranking factors in the final result set for different users. We perform user studies based on the information shared in the ego network of two of the authors.

We consider two queries for the user study: ‘budget’ and ‘privacy’. Both queries are selected because of their relevancy to a majority of users. The first author is labeled as ‘userA’ and the second author is labeled as ‘userB’. userA has 206 members in his ego network. The number of edges shared between the members is 1552, that is, an average of 8 edges per member. userB has 527 friends and the number of edges between the members are 1003, which results in an average of 2 members. It is evident from these statistics that the respective ego networks are very different in topological characteristics and our next step is to understand how the ranking factors impact the final result set formation. We compare the results based on how the value of each ranking factor holds up for each of the ranking factors. For example, we compare the degree value in the final result set as produced by each of the other three factors. The diversity factor, V.2, is analyzed by forming social groups using values of k from 1 to 6.

We start by comparing results based on ranking factor related to social context based factors. We plot the degree and the diversity value of a result set as computed by applying the different ranking algorithms. Next, we discuss the impact on time relevancy of the retrieved data based on different factors.

We plot the degree values in Figure 5a and Figure 5b for both queries for each user, respectively. We see that the value of the result set based on the ‘degree’ factor is the highest among all the result set values. This observation is quite intuitive because the purpose of the degree factor is to formulate results with only the users who share the highest number of mutual friends with the user performing the query. However, it is interesting to note the corresponding values for other factors and how they compare against the value of the degree factor. While the values are lower, we observe that the corresponding values are significantly lower for userB in comparision to userA. The value for ‘degree’ based factor for query *budget* is 181 and 211 for users userA and userB respectively. In comparison, for the time factor, the values are 140 and 23 respectively. The diversity-based values range between 104 and 178 for userA and between 50 and 70 for userB. We observe similar trends in value for the query *privacy* as well. We observe that, based on the factor considered during the ranking process, the degree values in the result set can vary significantly owing to the structural difference in the ego network of the user performing the query.



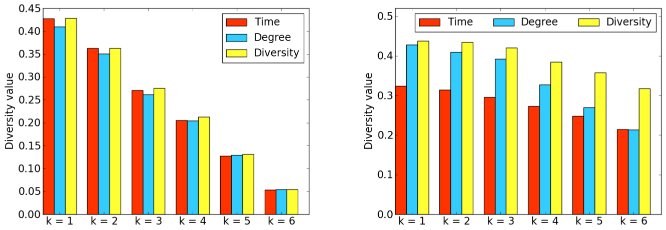
(a) userA

(b) userB

Figure 5: Query: budget: Degree analysis

Now we discuss the impact on the diversity value of the result sets. Values are plotted for each query in Figure 6 and Figure 7. The lowest diversity values are observed for result sets formed using the 'time' factor consistently under all conditions of k and for both the queries. We infer from this observation that information, once shared by a member in a social group, has a tendency to flow between the members of the particular social group before it is shared by members from other social groups. This leads us to conclude that result sets formed based on time of sharing can lead to information sources which originate within particular social groups and will have the lowest social diversity value. While the diversity based algorithm tries to maximize the value of social diversity in results, time factor among the other factors mostly retrieves results that have the least value of social context present.

We next analyze the difference in the values for each factor. For userA with the query *budget*, the maximum difference in values is 5% at $k = 1$. For other values of k, the difference in values is negligible. We observe similar trends in the values for the query *privacy* in Figure 7a. However, when we observe the values for userB, we see significant differences in the value between each result set in sharp contrast to the patterns seen for userA. Result sets formed based on diversity have the highest value amongst all the k values. For the query

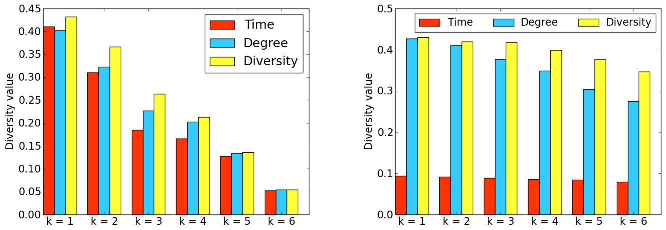


(a) userA

(b) userB

Figure 6: Query: budget: Diversity analysis

budget, we observe that the difference at various values of k is significantly less than the corresponding difference in values for query *privacy*. This is because many users from different social groups in the ego network of a user are interested in a generic query like *budget* and share relevant information. In contrast, for query *privacy*, information is shared by a selected few users inside a few social groups, upholding our previous conclusion that information flows inside particular social groups for a limited time before it spreads in the network. We next analyze the time property of the result sets.

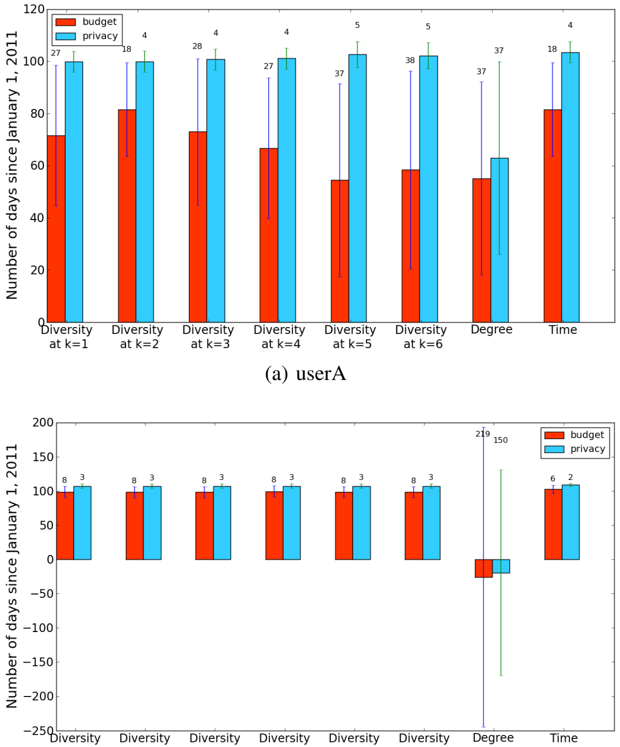


(a) userA

(b) userB

Figure 7: Query: privacy: Diversity analysis

We show the average and variation in the time stamp of the documents present in the final result sets formed in Figure 8a and Figure 8b for userA and userB, respectively. The time stamp results are plotted as the time since January 1, 2011. The 'time' factor results show the least amount of variation in the results. This is expected as the role of the 'time' factor is to select and rank the results based on the actual time when the results were shared without applying any social context factor. However, when we consider the other factors where input about the social relationship between the user performing the query and the source of information is taken into account, we see significant variation in the results retrieved for the final set. For userA, we observe that maximum variation occurs for the 'degree' factor at a variation of 37 days from the average of the time at which results were shared. The results are more striking for userB as we see results that are formed on the basis of 'degree' have a variation of 150 days and more. This shows that as we choose 'degree' of users as a ranking factor, final result set can contain results that were shared a significantly long time ago. In the next section, we conclude our work with a discussion about future work.



(a) userA

(b) userB

Figure 8: Time property analysis

IX. Concluding Remarks

In this work, we discuss our efforts to build *InfoSearch* over the Facebook platform as a prototype social search engine and provide scope to users to search through the posts shared by their friends. In the process, we identified three important factors related to ranking search results for social search systems. Users can employ either one of the factors to rank results as they search using *InfoSearch*. Based on data collected through the Facebook feeds of two users, we also performed user studies to understand the impact of ranking factors in the formation of result sets. We observed that 'time' based ranking of results, while providing the latest posts, fails to include sufficient social information in the result based on the value generated for both 'degree' and 'diversity' factors. Among the factors based on semantics of social relationships between a user performing a query and a user sharing a piece of information, 'diversity' based factor provides sufficient social context into the result set as well as performs well in comparison to 'degree' factor to include time characteristics in the result set. We believe the area of social search engines has an immense potential in the area of information search and retrieval and we want to expand this work into multiple directions. Firstly, we want to increase the impact of *InfoSearch* by inviting more users to use our system on a regular basis and provide us feedback on their opinion about the quality of results formulated. Secondly, we want to extend the system architecture to include the scope of distributed databases and develop the application into a distributed system capable of handling thousands of queries at any given time. Thirdly, we want to extend the factors involved in the ranking process to include other network based factors like 'betweenness', 'centrality' and 'interaction intensity between users'. Finally, we aim to design a proper methodology for evaluating social search engines.

References

[1] M. K. Mike Cassidy, "An update to google social search," http://googleblog.blogspot.com/2011/02/update-to-google-social-search.html, February 17, 2011.

[2] N. Wingfield, "Facebook, microsoft deepen search ties," http://online.wsj.com/article/SB10001424052748703421204576327600877796140.html, May 16, 2011.

[3] A. Borodin, G. O. Roberts, J. S. Rosenthal, and P. Tsaparas, "Link analysis ranking: algorithms, theory, and experiments," *ACM Transactions on Internet Technology*, vol. 5, no. 1, pp. 231–297, Feb. 2005.

[4] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine* 1," *Computer networks and ISDN systems*, vol. 30, no. 1-7, pp. 107–117, 1998.

[5] P. S. Dodds, R. Muhamad, and D. J. Watts, "An Experimental Study of Search in Global Social Networks," *Science*, vol. 301, no. August, pp. 827–829, 2003.

[6] L. Adamic and E. Adar, "How to search a social network," *Social Networks*, vol. 27, no. 3, pp. 187–203, Jul. 2005.

[7] T. Lappas, K. Liu, and E. Terzi, "Finding a team of experts in social networks," *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '09*, p. 467, 2009.

[8] J. Davitz, J. Yu, S. Basu, D. Gutelius, and A. Harris, "iLink: search and routing in social networks," in *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2007, pp. 931–940.

[9] A. Banerjee and S. Basu, "A social query model for decentralized search," in *Proceedings of the 2nd Workshop on Social Network Mining and Analysiss. ACM, New York*, vol. 124, 2008.

[10] D. Horowitz and S. D. Kamvar, "The anatomy of a large-scale social search engine," *Proceedings of the 19th international conference on World wide web - WWW '10*, p. 431, 2010.

[11] J. Haynes and I. Perisic, "Mapping search relevance to social networks," *Proceedings of the 3rd Workshop on Social Network Mining and Analysis - SNA-KDD '09*, vol. 09, pp. 1–7, 2009.

[12] A. Mislove, K. Gummadi, and P. Druschel, "Exploiting social networks for internet search," in *5th Workshop on Hot Topics in Networks (HotNets06)*. Citeseer, 2006, p. 79.

[13] R. Cross, A. Parker, and S. Borgatti, "A bird's-eye view: Using social network analysis to improve knowledge creation and sharing," *IBM Institute for Business Value*, 2002.

[14] D. Dhyani, W. K. Ng, and S. S. Bhowmick, "A survey of Web metrics," *ACM Computing Surveys*, vol. 34, no. 4, pp. 469–503, Dec. 2002.

[15] A. Plangprasopchok and K. Lerman, "Exploiting social annotation for automatic resource discovery," in *AAAI workshop on Information Integration from the Web*, 2007.

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

[17] S. Wasserman, *Social network analysis: Methods and applications*. Cambridge university press, 1994.

[18] Y. D. Network, "Term extraction documentation for yahoo! search," http://developer.yahoo.com/search/content/V1/termExtraction.html, June 15 2011.

[19] P. P. Index, "Content term extraction using pos tagging," http://pypi.python.org/pypi/topia.termextract/, June 15 2011.

[20] R. G. Vince Carey, Li Long, "Package 'rbgl," http://cran.r-project.org/web/packages/RBGL/RBGL.pdf, April 26 2011.