# Social Network Document Ranking

Liang Gou[1], Xiaolong (Luke) Zhang[1], Hung-Hsuan Chen[2], Jung-Hyun Kim[2], C. Lee Giles[1,2]
Information Sciences and Technology[1], Computer Science and Engineering[2]
The Pennsylvania State University, University Park, PA, 16802, USA
{lug129, xuz14, hhchen, jzk171}@psu.edu, giles@ist.psu.edu

## ABSTRACT
In search engines, ranking algorithms measure the importance and relevance of documents mainly based on the contents and relationships between documents. User attributes are usually not considered in ranking. This user-neutral approach, however, may not meet the diverse interests of users, who may demand different documents even with the same queries. To satisfy this need for more personalized ranking, we propose a ranking framework, Social Network Document Rank (SNDocRank), that considers both document contents and the relationship between a searcher and document owners in a social network. This method combines the traditional tf-idf ranking for document contents with our Multi-level Actor Similarity (MAS) algorithm to measure to what extent document owners and the searcher are structurally similar in a social network. We implemented our ranking method in a simulated video social network based on data extracted from YouTube and tested its effectiveness on video search. The results show that compared with the traditional ranking method like tf-idf, the SNDocRank algorithm returns more relevant documents. More specifically, a searcher can get significantly better results by being in a larger social network, having more friends, and being associated with larger local communities in a social network.

## Categories and Subject Descriptors
H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *relevance feedbacks, retrieval models, selection process*; H.1.2 [**Models and Principles**]: User/Machine Systems – *human factors.*

## General Terms
Algorithms, Experimentation, Human Factors.

## Keywords
Ranking, Social Networks, Information Retrieval, Multilevel Actor Similarity.

## 1. INTRODUCTION
A core component in search engines is the ranking algorithm which ranks the relevance of documents. Interest in improving the performance of ranking algorithms [1-5] will probably always continue. Some of these algorithms are user-neutral [4, 5] and measure the relevance of documents primarily based on the contents and relationships of documents. While this approach is effective in determining the relevance of documents to queries, they ignore one important factor involved in search, users who initiate queries. Users often have diverse interests and may want different documents, even when they use the same queries. A user-neutral ranking approach will not address this need because of the lack of user data in the ranking.

Personalized ranking algorithms have been proposed to include various types of user information [6] in ranking. To enhance ranking performance and improve search results, algorithms use such information as a user's search context [7], geographical location and searching histories [8], click-through logs [9], topics of interest [10] and personal bookmarks [11]. Some algorithms consider the information needs of a user's friends [12-14]. However, these algorithms largely focus on local activities of the user, and fail to embrace the large social contexts of the user.

In reality, users are involved in different social communities. Users are increasingly engaged in social networks through online services like Facebook, Flickr, and YouTube in order to communicate with their friends, family, and colleagues and share documents, images, and videos. We argue that their social networks may provide richer and more reliable clues about the purposes and interests of their information search. For example, when a devoted animal lover searches for information about snow leopards and uses a query "snow leopard", search engines often cannot tell whether required information is about an endangered species or an operating system and, as such, search results may not match the user's need. If the information about the user's social networks is available, search engines can disambiguate the query based on such information as which networks the user belongs to and who the user's friends are and then possibly deliver more relevant information.

We propose a new framework for personalized ranking. This framework, called Social Network Document Rank (SNDocRank), considers a searcher's social network when ranking the relevancy of documents. The premise of our methodology is that "birds of a feather flock together" [15]: 1) users tend to friend with those who share common interests, and 2) users are more interested in information from friends than from others. We also propose a Multi-level Actor Similarity (MAS) method to calculate actor similarity in social networks. By dividing a large social network into network modules at multiple scale levels, the MAS approach can dramatically accelerate the computation process.

The paper is organized as follows. Section 2 reviews related research. In Section 3, we present the SNDocRank framework and elaborate the MAS algorithm. In Section 4, we describe an experimental study on the effectiveness of the SNDocRank framework on information retrieval in a simulated video social network. After the discussion of our findings in Section 5, the paper concludes with future research directions.


*JCDL'10*, June 21–25, 2010, Gold Coast, Queensland, Australia.

## 2. RELATED WORK AND BACKGROUND

In general, two approaches can be considered to improve ranking algorithms in search engines: query dependent ranking (dynamic ranking) and query independent ranking (static ranking). Query dependent ranking algorithms estimate the similarity between the query term and the documents, and rank search results based on the similarity value. Many query-dependent models [16] have been proposed, including tf-idf [1] and BM25 [2]. The use of document metadata in ranking have also been proposed in [3], [17] A recent study shows that search results can be improved when ranking algorithms consider ever broader metadata, such as social annotations on documents made by other users [18].

Query independent ranking algorithms order search results based on the importance of documents. For documents on the Internet, the importance of a webpage is measured by algorithms based on the structure of the World Wide Web (e.g., PageRank [4], HITS [5], and their variations [19, 20]). In addition, users' click logs are employed as a measurement of the popularity of a document and can be an indicator of the document's importance [21].

Several studies have investigated approaches to offer personalized ranking to different users. A comprehensive review on personalized web search can be found in [6]. Here, we discuss two types of personalized search algorithms: personalization based on the information of the searcher and personalization based on the information of the searcher's peers.

Personalized web search based on a searcher's information relies on data associated with the searcher, such as search history, to adjust the weights of search results. Cubesvd [9] collected and analyzed clickthrough logs from MSN Messenger to improve search performance. Ucair [7] and Google's personalized search [8] use search history to customize ranking. In [10], a user first specifies a topic of interest and the distance between the user's interest vector and the ODP (Open Directory Project) directories is calculated to determine whether the document interest. Hyperlink based personalization [11] collects a user's bookmark or frequently visited page sets as hubs and re-ranks search results when a query is submitted.

Information provided by peers or embedded in other social structures can also be used to improve ranking. FolkRank [22] leverages the structure of folksonomy and finds the communities to restructure search results. In [12], the history of the other users with similar tastes to those of the searcher is analyzed for ranking. When the searcher explicitly specifies friends, web pages previously visited by friends are assigned higher ranking weights [13, 14]. Other research explored methods to infer friend relationships when friend information is not provided. In [23], a *top-k* algorithm was proposed to improve search results by considering social expansion (e.g., the strength of relationship among users) and semantic expansion (e.g., the relevance of tags). In [24], the distance between two nodes and the clustering information in social networks are used to improve search results.

In summary, current personalized search methods are limited to integrating information about the visiting and searching histories of the searcher or direct friends, and do not consider the searcher's larger social contexts, such as the interests of those who are not direct friends, but are connected through others.

## 3. SNDocRank FRAMEWORK

To further improve personalized ranking, we propose a new ranking framework, SNDocRank, to consider both the relevance of documents and the relationships between the searcher and others in a social network. In this section, we first introduce the framework, and then discuss our multi-level actor similarity (MAS) method to accelerate the computation of actor similarity. In Section 3.3, we describe the implementation of the SNDocRank framework and MAS method in document ranking. Our focus here is on the general concept and ideas of SNDocRank. Mathematical details of our framework and algorithms are available in other work [25].

### 3.1 Framework of SNDocRank

Figure 1 illustrates major components of the SNDocRank framework. The framework has three core components: an actor similarity module to compute actor similarity scores, a document matching module to match user queries with indexed documents, and a SNDocRank module to produce the final ranking by combining document relevance scores with actor similarity scores. The document matching module is a typical term-based search engine. Thus, in this section, we discuss the actor similarity module and the implementation of the SNDocRank module.

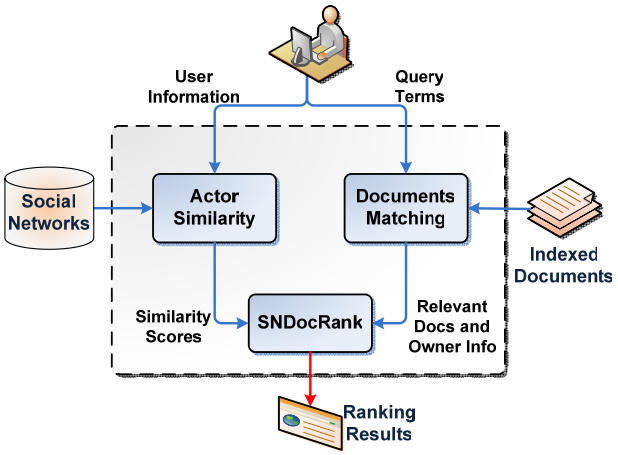



**Figure 1. SNDocRank Framework.**

### 3.2 Multi-Level Actor Similarity (MAS)

One way to expand the scope of social relationships in a social network for a personalized searcher is to consider actor similarity in ranking algorithms. The actor similarity of social networks measures how similar two actors in a social network are to each other based on the structural information of the social network. Several approaches are introduced in [26]. In this section, we introduce the concept of actor similarity and our MAS algorithm.

#### *3.2.1 Actor Similarity Algorithms*

One of the simplest ways to measure actor similarity in social works is cosine similarity based on structural equivalence [27]. In this approach, two actors are regarded similar if they share many neighbors in a social network. A basic method is cosine similarity which only considers directly connected actors as neighbors, but fails to integrate larger social contexts of a social network.

LHN (Leicht, Holme, Newman) vertex similarity [28] improves the similarity measure by considering the similarity scores of the whole neighborhood, rather than direct connection. For two nodes in a network, if their immediate neighbors are similar, then these two nodes are similar, even though they are not directly connected. The calculation of the similarity of two nodes with the LHN Vertex similarity is a recursive process, which involves not

only their direct neighbors but also all nodes connected to the neighbors. Consequently, the similarity score of the LHN method integrates both the local connectivity of these two nodes (e.g., the direct connections) and their global connectivity (e.g., the nodes indirectly linked to them).

However, the LHN vertex similarity approach faces one challenge: scalability. Because of the recursive process that involves expensive matrix multiplication, its computational complexity is extremely high. It becomes impractical in processing large social networks with thousands or millions of nodes and edges.

### *3.2.2 MAS Approach for Actor Similarity*

To address the issue of computation complexity, we developed an approach that first clusters a large social network into smaller ones at multiple scale levels and then computes the similarity within and between network clusters. This hierarchical approach, called Multi-level Actor Similarity (MAS), preserves the primary benefit of the LHN vertex similarity, which is the availability of the global structural information in similarity scores, and at the same time, significantly reduces computation complexity, making our SNDocRank approach a feasible algorithm for various applications.

Our MAS approach includes three steps. The first step is an algorithm to cluster and aggregate a social network at multiple levels and create a node cluster hierarchy. Each network node can belong to one and only one node cluster in the hierarchy. The network of node clusters at each hierarchical level captures the main structural characteristics of the network and serves as a backbone of the network at that level. Then, a weighted LHN vertex similarity method is applied to compute the similarity among these node clusters in the hierarchy. The similarity between two node clusters in the backbone network offers contextual information for the similarity between network nodes within them. Finally, the similarity of any two network nodes is computed by considering the similarity between each node and its parent node cluster, and the similarity between their parent node clusters.

Figure 2 shows an example of computing the similarity of two nodes in a social network with our MAS method. Assume the goal is to calculate the similarity between Node 5 and Node 9 (Figure 2a). The first step of our algorithm is to generate a hierarchy of node clusters based on the structural characteristics of the network. Assume three node clusters can be identified (Figure 2b) and networks and node clusters form a hierarchy (Figure 2c). Then, the whole network can be simplified and be represented as a backbone network with these three node clusters, $C_1$, $C_2$, and $C_3$. Each cluster contains aggregated information of its member nodes, and the relationship among them is also the aggregation of the relationships among their members (Figure 2d). The similarity values among these node clusters can be calculated by applying the LHN method on the backbone network. Then, the similarity values between Nodes 5 and 9 is calculated by considering the similarity values between Node 5 and its parent cluster $C_2$ ($S_{5C2}$), between Node 9 and its parent cluster $C_3$ ($S_{9C3}$), and between two clusters $C_2$ and $C_3$ ($S_{C23}$).

#### *3.2.2.1 Multi-level Node Clustering and Aggregation*

Social network clustering, or community detection, is a continuing topic of research in social networks. Various algorithms have been proposed, such as the fast community detection algorithm by

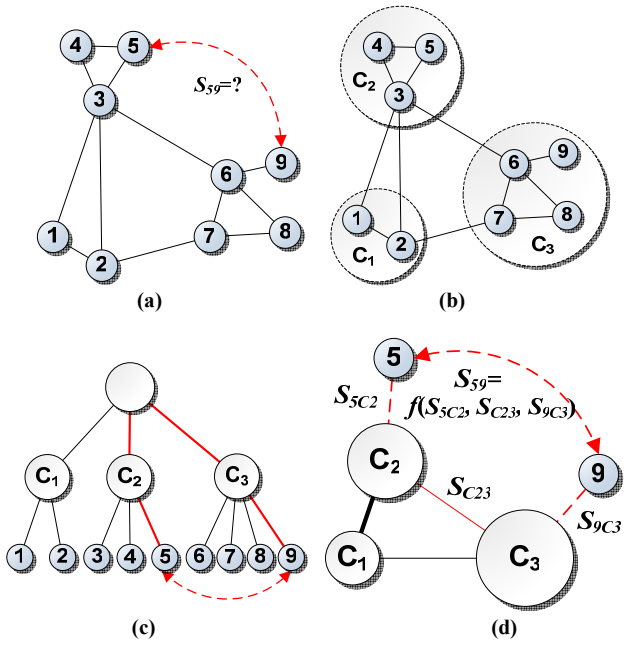



**Figure 2. An Example of Multi-level Actor Similarity.**

Clauset, et al. [29]. This algorithm is based on a quality measurement for clustering a network: ***modularity*** [30]. A good clustering of a network, which has high value of modality, maximizes the number of edges within clusters and minimizes the number of edges between clusters. Clauset, et al. [29] provides more details about the measure of modularity. Our clustering method is based on this modularity-oriented approach.

The result of the clustering process is a hierarchical tree, in which each leaf node is an actor, and each non-leaf node is a group of actors. One of the concerns of this clustering process is that the output is an unbalanced tree, in which some clusters are very large while some are very small. The unbalanced tree could have serious computational effects, because large clusters may become a scaling bottleneck for the whole process. One way to address this issue is to set a size limit for node clusters so that large clusters are broken into smaller ones.

Aggregation is an approach to describe the main structural features of a node cluster. After aggregation, a node cluster can be treated as a network node, and the edges between the node cluster and another node (network node or node cluster) can also be grouped as one single meaningful connection.

Various methods can be applied to node aggregation and edge aggregation. In our implementation, we use the node with highest degree in a cluster to represent the whole cluster, and add the edges between a node cluster and another node as the aggregated edge that now indicates the connection strength of two entities.

The result of network clustering and aggregation is a weighted hierarchy of network nodes and clusters. In the hierarchy, leaf-nodes are the nodes from the original social network. Non-leaf nodes are the clusters that contain nodes that share some common structural characteristics. As the result of edge aggregation, the clustered networks are weighted. The clustered networks preserve the key structural features of the social network and the structure at different levels of hierarchy serves as a backbone for further network analysis.

### 3.2.2.2 Weighted LHN Vertex Similarity

To compute the node similarity in the hierarchy, we need to extend the LHN vertex similarity method so that it can be applied to weighted networks. The LHN vertex similarity algorithm uses an adjacency matrix to describe the connectivity of network nodes. The values in the matrix can be binary, either 1 or 0, which indicates that two nodes are connected or not. For our network hierarchy, the values of the adjacency matrix are no longer binary because of the weighted edges.

One way to apply the LHN vertex similarity algorithm to a weighted network is to normalize its weighted values. The rationale of this approach can be derived from the similarity concept in LHN. The similarity in LHN is the neighbor similarity plus the self-similarity defined as:

$$S_{ij} = \phi \sum_{v} A_{iv} S_{vj} + \varphi \delta_{ij} \quad (1)$$

where $S_{ij}$ is the similarity of vertex $i$ to vertex $j$; $\delta$ and $\varphi$ are control parameters; $\delta_{ij}$ equals to 1 when $i=j$, otherwise 0, and $A_{ij}$ is an element in the adjacency matrix $A$.

The values in $A$ in the LHN vertex similarity can be regarded as the connectivity of a pair of network vertices. The aggregated value of an edge between two node clusters in the clustered network hierarchy represents the strength of the two clusters. Thus, the normalized value can be used to represent their similarity weight with their neighbors.

Therefore, the similarity matrix $S$ of the hierarchy of network clusters can be iteratively computed as:

$$DSD = \frac{\alpha}{\lambda_1} N(A_w)(DSD) + I \quad (2)$$

where $S$ is the similarity matrix; $A_w$ is a weighted adjacency matrix; $\alpha$ is a constant; $\lambda_1$ is the largest eigenvalue of $A$; $N(\bullet)$ is a normalized function; and $D$ is the diagonal matrix, in which the diagonal elements are the degrees of individual vertices.

$DSD$ can be easily computed iteratively by initially setting $S$ as 0. With a proper $\alpha$, this formula can converge quickly after 100 iterations or less [28]. After $DSD$ is calculated, the similarity matrix S can be obtained by:

$$S=D^{-1}(DSD)D^{-1}. \quad (3)$$

### 3.2.2.3 Actor Similarity

With the similarity matrix $S$ of the hierarchy of network clusters determined, the similarity between any pair of actors in the social network can be easily calculated. If two actors are in the same cluster, their similarity can be directly calculated by the LHN algorithm within the cluster. Because the size of an individual cluster is small, this computation is scalable for large networks.

Computing the similarity between two actors in indifferent clusters is more complicated. It involves the similarity values between each actor and its parent cluster, and the similarity between two parent node clusters, as shown in Figure 2.

### 3.2.2.4 Algorithm Comparison

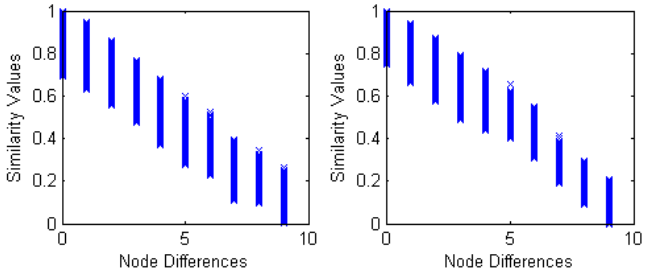
We evaluated the weighted LHN and MAS methods by comparing the network similarity results between LHN and weighted LHN, as well as between LHN and MAS with a similar approach used in [28]. We generated a fully connected social network with 1000 nodes, in which each node was assigned a random integer from 0 to 9 as an attribute value and the edges among nodes were created with this probability:

$$P(\Delta t) = p_0 e^{-a\Delta t} \quad (4)$$

where $\Delta t$ is the difference of the attribute value of two nodes and a measure their similarity, and $\alpha$ and $p_0$ are parameters to control the distribution shape (for this case 2.0 and 0.12 respectively).

### *Comparison of LHN and Weighted LHN*

We first used LHN to calculate the similarity values with the unweighted social network generated by the model defined in Equation 4. Then, we assigned edges in the network with weights, $w=1/(\Delta t+1)$. This weight indicates that nodes with smaller differences of attribute values have larger edge strength. We applied the weighted LHN to this new network.



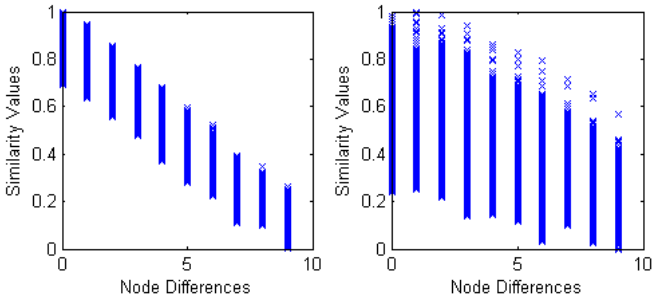

**(a) LHN.** **(b) Weighted LHN.**

**Figure 3. Scatter Plots of LHN and Weighted LHN**

Figure 3 shows the scatter plots of similarity results by LHN and weighted LHN for all node pairs against the node differences in the network. The similarity values are logarithmic and rescaled from 0 to 1 for comparison. As shown, the weighted LHN has a sharper slope and narrower range of similarity values for each node difference than the standard LHN does. We believe this indicates that the weighted LHN method generates more accurate results. This may be due to the fact that weighted edges between nodes provide more information to what extent two nodes are actually similar.

### *Comparison of LHN and MAS*

We also compared the LHN method and our MAS algorithm. Figure 4 illustrates the scatter plots of the similarity values by two algorithms. As shown, the trends of the LHN and MAS plots are comparable, although the slope trend of MAS is not as sharp as that of LHN and the range of similarity is not as wide as LHN. This implies the accuracy of the LHN approach is better than that of the MAS.




**(a) LHN Similarity.** **(b) MAS.**

**Figure 4. Node Similarities with LHN and MAS.**

But by sacrificing accuracy, our MAS approach gains speed. MAS can dramatically improve the computation efficiency of network similarity. The complexity of the LHN method under the Coppersmith–Winograd algorithm is $O(n^{2.376})$ [31]. The complexity of our MAS method is much better: $O(n^{1.688}\log n)$ [25], which makes this approach quite scalable for large $n$.

## 3.3 Implementation of SNDocRank

SNDocRank considers both the relevance of documents to a query and the similarity between document owners and a searcher. Thus, a composite ranking can be written as:

$$SND(u, q, d) = f(R(q, d), S(u, o)) \quad (5)$$

where $SND(u, q, d)$ is the ranking score of a document $d$ when a user $u$ is conducting a search with query $q$; $R(q, d)$ is the value that indicates the relevance of $d$ to $q$ based on a term-document similarity function, such as tf-idf [1] and BM25 [2]; and $S(u, o)$ is the similarity value between the user $u$ and the owner of the document $o$. Both $u$ and $o$ are in the same social network.

The implementation of SNDocRank is flexible. Different document similarity functions (e.g., tf-idf or BM25) can be used. The similarity component can also be selected based on the size and nature of social networks. For example, for smaller social networks, LHN based algorithms can be directly used, because the computation complexity maybe tractable. Or for a shallow social network that does not have many layers, cosine similarity may be good enough. For complex networks, the MAS approach can be used. Furthermore, the format of the function $f(\bullet)$, which combines document similarity and actor similarity, can also be adaptive based on the topological features of social networks and the nature of document collections.

# 4. EXPERIMENT AND RESULTS

Consider the following experiment to evaluate the SNDocRank framework and the MAS method. In this section, we first introduce the datasets used in the evaluation. Then, we present the evaluation study and its results.

## 4.1 Datasets

### *4.1.1 Data Acquisition*

Our study requires two sets of data: documents and social networks. We explored various options for datasets, and eventually decided to use data from YouTube.com, which offers both rich document data (video) and extensive social network information. We did not use data from popular social network services, such as Facebook.com, because the datasets in these sites tend to be rich on social networks, but poor on documents. Information sharing services like Flickr.com can offer data similar to that of YouTube.com, but the access to the data in such services is more restricted, compared to that in YouTube.com.

We adopted a strategy of breath-first search (BFS) to crawl the social network data and associated video metadata in YouTube.com. We first started with five seed users randomly chosen with high degree and different interests, and obtained their friends and video information. Then we used these friends as the new centers and fetched the friends and video metadata from these centers. This process was iterative and stopped until no more targets were available or the number of retrieved users exceeded a pre-defined value.

### *4.1.2 Document Data*

In the YouTube dataset, each document contains rich metadata of videos, such as title, genre, tags, description, uploader, published time, url, rating, etc. These metadata were downloaded with our crawler and stored locally. We did not download the videos, however. This was because our document indexer is still text-based and video contents are currently not of interest. Also, users can directly access the videos via the urls , making it unnecessary to store videos locally.

The videos are labeled with 15 main categories (music, entertainment, sports, education etc.) provided by YouTube. In this experiment, we defined a searcher's interest with one of these video categories. If the majority of a searcher's videos fall into one category, we regard that category as the searcher's interest. We assume that a searcher only has one dominant interest group which is defined by the main category of the searcher's videos, although in practice the searcher may have several interests at the same time.

Table 1 summarizes the data attributes of video documents we downloaded and presents the attributes values of an example video document. The name of the uploader of the video and the url are anonymized.

**Table 1. Video attributes and values.**

| *Attribute* | *Value* |
|---|---|
| Title | Naive Gaussian Elimination Method |
| Uploader | *** ****** |
| Genre | Education |
| Tags | numerical, methods, Gaussian, elimination, numericalmethodsguy |
| Description | We are trying to record lectures with Camtasia and a Smart Monitor in our offices. This is a sample video of Gaussian Elimination with Partial Pivoting |
| URL | http://www.youtube.com/watch?v=**** |
| Published | 2009-08-06T00:37:44.000Z |
| Rating | 5.0 |

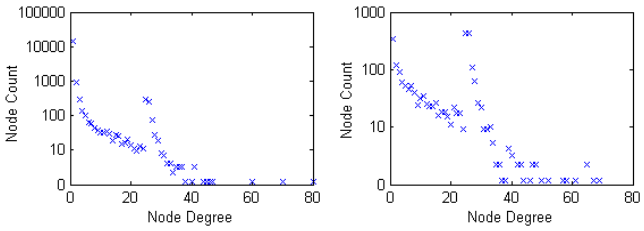
With the metadata of video documents, we built the indexes for video documents. The fields we used in the metadata include title, tags, genre and description. The values in these fields were parsed into terms to create an inverted index using Solr/Lucene, in which each term in a specific field points to a collection of video documents. These indexes can be easily used to generate term-document similarity scores.

### *4.1.3 Social Network Data*

In YouTube.com, each user has attributes such as user name, friends, videos uploaded, age, gender, hometown, hobby, and about-me. Social networks can then be constructed based on the friend information obtained. An edge was created between two users if they added each other as a friend.

In this experiment, we used two fully connected social networks based on the data we downloaded:

- Network A: a larger social network that consists of 16,576 different registered users and 39,281 videos uploaded by these users;
- Network B: a smaller network that has 2,264 users and 7,309 videos.




**(a) The social network A** **(b) The social network B**

**Figure 5. Degree Distribution of Two YouTube Social Networks.**

Network A has 42,429 edges. Its maximum degree is 89, and the degree mean is 5.11. Network B has 6038 edges with the maximum degree of 69 and the degree mean of 5.33. The degree distributions of two networks are shown on a log scale in Figure 5. Both figures roughly follow a power law distribution and have the scale-free feature of large social networks [33], except that both networks have a spike of frequency at degree of around 25. This spike is more obvious in Network B (Figure 5b), which may result from its smaller number of users. Users with degree of 25 are typical in these networks and the value is a result of download limitations.

The distributions of user interest categories in two networks are shown in Figure 6. The interest categories in two networks follow a similar pattern: the dominant interest group is music, with 43.35% of users in Network A and 41.67% of users in Network B. Entertainment and comedy interest groups are ranked second and third.

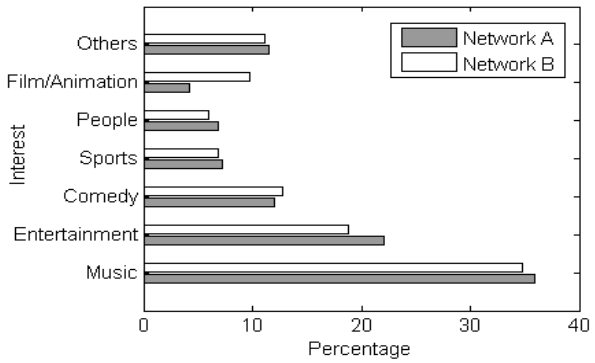



**Figure 6. Distributions of User Interests in Two Networks.**

## 4.2 Video Search with SNDocRank

We implemented the SNDocRank framework in a mobile video social network application [32] and conducted experiments on information retrieval.

With the MAS algorithm, we calculated the similarity values of the pairs of actors in the two social networks. Figure 7 shows the distribution of MAS values among a user and other users in the social network A. The similarity values are plotted as a function of the log of MAS and normalized to a range from 0 to 9. These values follow a normal distribution: the mean close to 5 with a small proportion with high and low values. The tail on the right indicates that the MAS values can effectively distinguish between users with high similarity values and those in the middle.

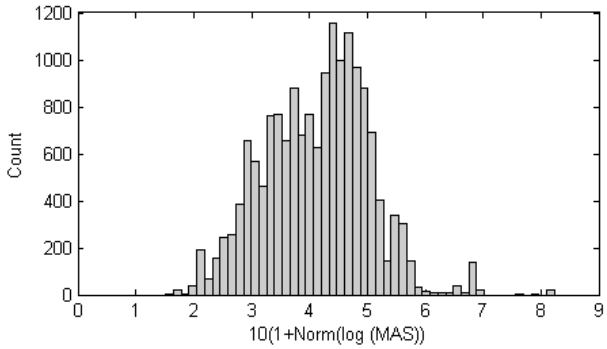



**Figure 7. The Distribution of MAS Values Across a User and Other Users in Social Network A.**

## 4.3 Evaluation Setup

### *4.3.1 Experiment Procedure*

We first selected a set of users from two networks whom we assumed needed to search for videos in YouTube. Our assumption was that a returned result is good for a searcher only when the result is both relevant and interesting. Recall that the interest of a searcher is defined by the dominant category of videos that the searcher had uploaded.

Table 2 shows the searchers we selected from two networks. From the larger social network A, we chose 3 categories of interest: music, sports, and animation. Within each category, we selected six users at three levels of degree—high (H), medium (M), and low (L), with two users at each level. The dominant category of interest of a searcher was determined by the ratio of videos in a category over all videos. For the smaller network B, because of its size, we only chose one category—music, which is the largest category. Six searchers at three levels of degree were selected, again two at each level.

**Table 2. Searchers Selected for the Experiment.**

| *Network* | *Interest Category* | *Degree* | *Searcher 1: Video Ratio, Degree* | *Searcher 2: Video Ratio, Degree* |
|---|---|---|---|---|
| A | Music | H | 0.73, 63 | 0.92, 44 |
| | | M | 0.80, 25 | 0.76, 27 |
| | | L | 1.00, 8 | 0.83, 7 |
| | Sports | H | 0.70, 25 | 0.72, 27 |
| | | M | 0.72, 17 | 0.67, 11 |
| | | L | 0.81, 4 | 0.74, 2 |
| | Animation | H | 0.80, 27 | 0.94, 29 |
| | | M | 0.67, 15 | 0.52, 11 |
| | | L | 0.72, 5 | 0.52, 2 |
| B | Music | H | 0.75, 61 | 0.87, 50 |
| | | M | 0.94, 26 | 0.76, 27 |
| | | L | 0.71, 2 | 0.83, 7 |

To examine the effectiveness of the proposed composite ranking algorithm, we compared following three algorithms:

- **Baseline**: tf-idf [1] was used as the base line method.
- **Cosine**: this method integrates the base line and the cosine similarity in social networks using a product:

$$SNDoc(v, t_i, d_j) = tf_{i,j} \cdot idf_i (1 + \rho Cos_{vu}) \quad (6)$$

where $Cos_{vu}$ is the cosine similarity value between user $v$ and $u$, and parameter $\rho$ is tuned at 0.3. Both tf-idf and cosine values are normalized.

- **MAS**: This algorithm combines the base line and our MAS method in a product:

$$SNDoc(v, t_i, d_j) = tf_{i,j} \cdot idf_i \cdot (1 + \rho Norm(\log MAS_{vu})) \quad (7)$$

where $MAS_{vu}$ is the actor similarity value between user $v$ and $u$ using the MAS method. *MAS* are non-zero values. $Norm(\bullet)$ normalizes the values, and the parameter $\rho$ is 0.5.

In each category, we choose 15 queries that are related to the category but at the same time could also refer to things that do not belong to the category. Queries in the music category were about bands, song titles, etc.; queries in the sports category included the names of sports clubs, sports stars, etc.; and queries in the animation category had cartoon themes, cartoon titles, etc. Table 3 shows the queries we used in the test. We conducted searching under each searcher for all theses queries.

**Table 3. Queries Used in Evaluation.**

| *Category* | *Queries* |
|---|---|
| Music | angel, bear, forest, friends, graduate, heart, jean, ocean, rainbow, red, sea, sky, star, summer, wind |
| Sports | basketball, dolphins, fan, heat, Henry, highlight, impact, kings, match, Milan Italy, football, Suzuki, Tiger, water sports, Yamaha |
| Animation | cowboy, death note, Disney, dragon, fighter, hunter, metal, ninja, Prince, Princess, sailor, spider, Spiderman, super, transformer |

The top twenty returned videos of each query were mixed and presented to three PhD students for independent evaluation. All of these students are regular YouTube users. They evaluated each returned result based on to what extent the returned video was relevant to a query as well as to what extent the video was related to the searcher's interest. Table 4 shows the scores used to rate the relevance of a return. The highest score of a return is 3 and the lowest score is 0. After a pilot experiment, the inter-rater reliability among three raters was 68.2%, indicating that they reached a reasonable agreement about the relevance criteria [34].

**Table 4. Relevance Evaluation Table.**

| | | Content Relevance | | |
|---|---|---|---|---|
| | | High | Medium | Low |
| Interest | High | **3** | **2** | **0** |
| | Medium | **2** | **1** | **0** |
| | Low | **0** | **0** | **0** |

### *4.3.2 Evaluation Metrics*

We used a popular metric, *Normalized Discounted Cumulative Gain (NDCG)* [35], to evaluate the ranking algorithms. The NDCG method measures the usefulness of the ranking result based on the relationship between the relevance scale of documents and the document's position in the ranking. The premise of DCG is that highly relevant documents are more useful when they have higher ranks in the result list. The NDCG at position $k$ is given by:

$$NDCG_k = \frac{DCG_k^R}{DCG_k^T} \quad (8)$$

and

$$DCG_k^X = Rel_1(X) + \sum_{i=2}^{k} \frac{Rel_i(X)}{\log i} \quad (9)$$

where $Rel_i(X)$ shows the level of relevancy for the result at position $i$ in rank X. $DCG_k^T$ is the value for the optimal rank at position $k$. In our experiment, we have four levels of $Rel_i$, as shown in Table 4.

## 4.4 Results

### *4.4.1 Overall Performance*

We first evaluated the performance of three different ranking approaches. Figure 8 show the NDCG results of three approaches. These NDCG values were averaged over all queries without considering the differences in the size of social networks, the degrees, and interests of searchers. As shown, both SNDocRank methods (MAS and cosine) perform better than the baseline algorithm. The MAS method is most effective. For the first 20 search results, the NDCG values of MAS are 10 points higher than that of the baseline algorithm (about 20% relative

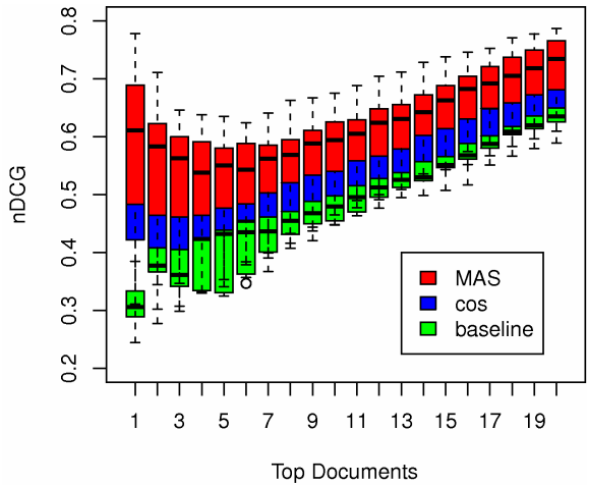
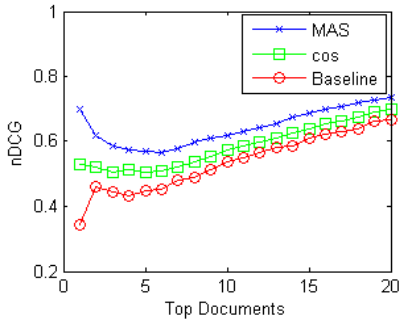



**Figure 8. NDCG for Three Algorithms over All Queries.**

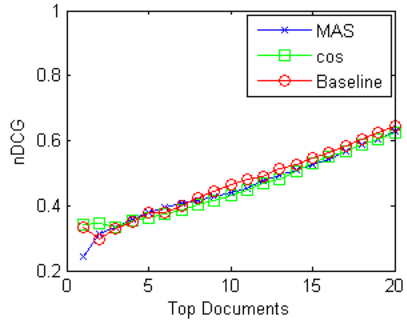
improvements) and 5 points higher than that of the cosine method (about 10% relative improvements).

### *4.4.2 Effect of Network Size*

We also examined the performance of three algorithms in two networks (large and small). To make the results more comparable, we only used searchers for the music category in both networks.




**(a) Larger Social Network A.**




**(b) Smaller Social Network B.**

**Figure 9. Average NDCG for Users in Two Networks.**

The NDCG results from two social networks are shown in Figure 9. The NDCG scores are averaged over all queries from searchers of music for the two networks. As shown, for the larger social network A, the SNDocRank (both the MAS and cosine methods) performs better than the baseline algorithm, and the MAS method is the best. However, the difference among three methods in the smaller social network B is minimal.

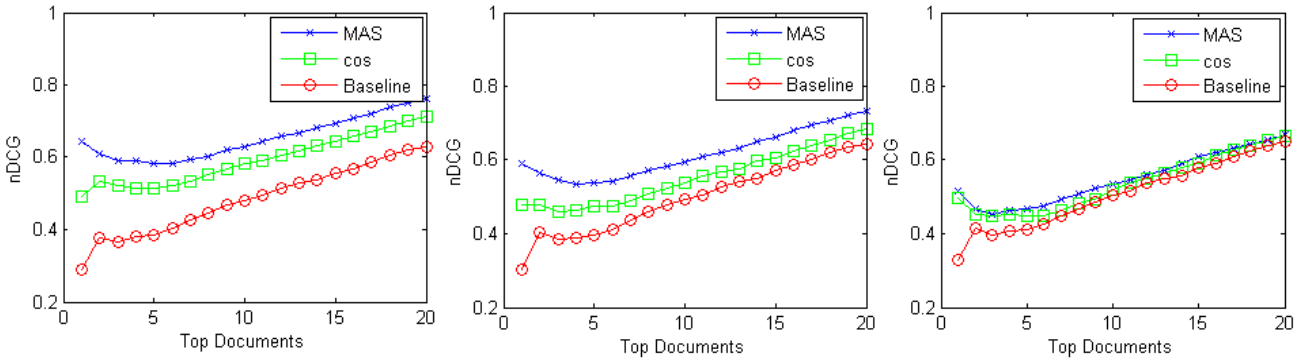
### *4.4.3 Effect of Degree*

We also tested the three algorithms over searchers with three types of degree in the larger social network A. Figure 10 shows the NDCG results. The NDCG results are averaged over the searchers with the same type of degree.

In general, the NDCG values from the MAS method are higher than those of the baseline and cosine approaches across all measures of degree. For high-degree searchers, the NDCG values from MAS are 15 points higher than the values from the baseline method (over 25% relative improvement) and 5 points higher than the values from the cosine method (about 10% relative improvement). However, when the degree goes down, the NDCG values of both MAS and Cosine drop. For low-degree searchers, the NDCG results from three algorithm are very close.

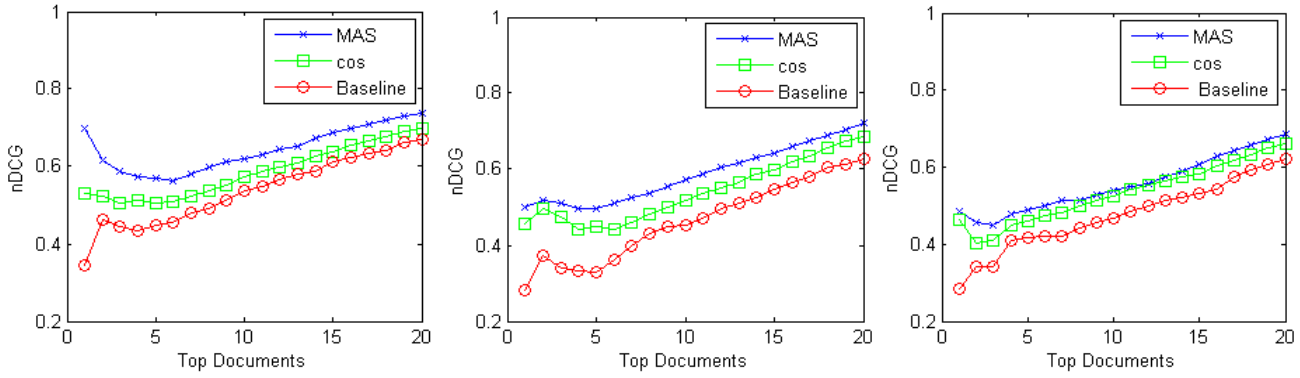
### *4.4.4 Effect of the Size of Interest Groups*

Our last comparison was on the effect of the size of interest groups in a social network based on search results. We chose for comparison three interest groups from the larger social network A - music, sports, and animation. The sizes of these three groups were different (see Figure 6). Figure 11 shows the NDCG results




**(a) High Degree Searchers** **(b) Medium Degree Searchers** **(c) Low Degree Searchers**

**Figure 10. NDCG Results for Searchers with Different Degrees.**




**(a) Music Searchers (43.35%)** **(b) Sports Searchers (10.16%)** **(c) Animation Searchers (3.87%)**

**Figure 11. NDCG Results for Searchers in Different Interest Groups.**

for searchers. The results are averaged over all queries with the same type of interest.

From Figure 11, we can see that the MAS approach outperforms both the baseline and cosine methods regardless of the size of an interest group. The advantage of the MAS approach is more prominent for the two large interest groups of music and sports.

However, the magnitudes of improvement vary from group to troup. For the largest interest group, music, which was 43.35% of the total network, the NDCG values from the MAS method are 8 points higher than the values of the baseline method (over 12% relative improvements) and 4 points higher than the values of the cosine method (about 5% relative improvement). For the smallest interest group, Animation (3.87% of total network), the NDCG results from MAS and cosine algorithms are very close but both are still better than the baseline.

## 5. Discussion

The results of our evaluation studies indicate that overall, the SNDocRank framework can return better search results than the traditional tf-idf ranking algorithm in terms of relevance, the matching of interests with searchers, and the ranking effectiveness of returned results. Compared with the cosine similarity method, our multi-level actor similarity method outperforms the cosine similarity algorithm consistently across different sizes of social networks, different degrees of searchers, and different sizes of interest groups in a social network. This indicates that the structure of a searcher's social network can provide clues about the user's information needs and then can be used to help improve the performance of ranking algorithms.

Our results also indicate the sensitivity of our SNDocRank approach to certain characteristics of a searcher's social network. As shown in Figures 9, 10 and 11, the search results from the SNDocRank method (both MAS and cosine) vary with the size of a searcher's social network, a searcher's degree, and the size of a searcher community in a social network. Although the SNDocRank method considers the global information of a social network, it becomes effective only as the size of a network reaches a certain magnitude. We believe this problem is similar to that of the cold-start problem in collaborative filtering [36], which for good performance needs sufficient information about new users.

The degree of a searcher in a social network can also affect the performance of the SNDocRank framework. Generally speaking, both MAS and cosine methods benefit high-degree searchers more than they do low-degree searchers. This may be due to the fact that higher-degree searchers leave more clues and traces about themselves in the social network, which can then be used to improve document rankings. Our MAS method is more effective than the cosine approach because MAS considers the global information of social networks, while the cosine approach only focuses on that from direct neighbors.

The size of local communities in a social network also affects the SNDocRank results. Both MAS and cosine algorithms favor larger interest groups. This may again be related to the availability of information about searchers. Larger communities tend to spread information about themselves and their members more broadly than smaller communities.

Our results suggest actions that users can pursue to improve searching results, at least for our methods. First, a user should join large social networks, because our SNDocRank methods benefit more from large social networks than from small networks. Second, a user should try to be connected as many people as possible to increase their degree, which for our method leads to better search results. Finally, in a social network, a user should be connected to large communities or interest groups.

We should also be aware of the potential problems of the SNDocRank method. This framework benefits large social networks more, favors those well-connected people, and promotes large and popular local communities more in a social network. In the long run, these biases in ranking documents may lead to unexpected social consequences in information dissemination.

## 6. CONCLUSION AND FUTURE WORK

In this paper, we presented SNDocRank, a framework that incorporates both the term-document similarity and the actor similarity in associated social networks to rank search results to fit the interests of searchers. To deal with the complexity of similarity computation in large social networks, we developed a multi-level actor similarity (MAS) method for the SNDocRank framework. The results of our experiments in a simulated social network and YouTube video search show that our approach returns more relevant information than those ranking methods not considering the broad structural information of social networks. Our experiments also offer searchers recommendations on how to obtain more relevant search results within their social networks.

For new directions, we are interested in applying the SNDocRank method to text document search and image search, and examining the effectiveness of the SNDocRank approach on different types of social networks. In addition, one could investigate the SNDocRank framework by considering other advanced ranking approaches, such as PageRank and HITS. Finally, the SNDocRank function could be improved by using machine learning techniques to determine parameters that could balance the impact of social actor similarity and term-document similarity on ranking results.

## 7. ACKNOWLEDGEMENTS

Part of this work was funded by Alcatel-Lucent and NSF.

## 8. REFERENCES

[1] G. Salton & C. Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management.* Vol. 24, No. 5, pp. 513-523, 1988.

[2] K. S. Jones, S. Walker & S. E. Robertson. A probabilistic model of information retrieval: Development and comparative experiments. *Information Processing and Management.* Vol. 36, No. 6, pp. 779-808, 2000.

[3] N. Craswell, D. Hawking, & S. Robertson. Effective site finding using link anchor information. In *Proceedings of the ACM Conference on Information Retrieval (SIGIR '01)*, pp. 250-257, 2001.

[4] S. Brin & L. Page. The anatomy of a large-scale hypertextual Web search engine, In *Proceedings of the International World Wide Web Conference (WWW '98)*, pp. 107-117, 1998.

[5] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*. Vol. 46, No. 5, pp. 604-632, 1999.

[6] A. Micarelli, F. Gasparetti, F. Sciarrone, & S. Gauch Personalized search on the World Wide Web. In *Lecture*

*Notes in Computer Science.* Volume 4321/2007 (the Adaptive Web), pp. 195-230, 2007.

[7] X. Shen., B.Tan, & C. Zhai, C. Ucair: Capturing and exploiting context for personalized search. In *Proceedings of the Information Retrieval in Context Workshop, SIGIR IRiX'05*, 2005.

[8] Google. *Personalized search for everyone*. http://googleblog.blogspot.com/2009/12/personalized-search-for-everyone.html. Last retrieved on April, 16, 2010.

[9] J.-T. Sun, H.-J. Zeng, H. Liu, Y. Lu & Z. Chen. Cubesvd: A novel approach to personalized web search. In *Proceedings of the International World Wide Web Conference (WWW '05)*, pp. 382-390, 2005.

[10] P. A. Chirita, W. Nejdl, R. Paiu & C. Kohlschütter. Using odp metadata to personalize search. In *Proceedings of the ACM Conference on Information Retrieval (SIGIR '01)*, pp. 250-257, 2001.

[11] G. Jeh & J. Widom. Scaling personalized web search. In *Proceedings of the International World Wide Web Conference (WWW '03)*, pp. 271-279, 2003.

[12] M. Montaner, B. López, & J. L. de La Rosa. A taxonomy of recommender agents on the internet. *Artificial Intelligence Review*. Vol. 19, No. 4, pp. 285-330, 2003.

[13] A. Mislove, K. P. Gummadi & P. Druschel. Exploiting social networks for internet search. In *Proceedings of the 5th Workshop on Hot Topics in Networks (HotNets'06)*, 2006.

[14] M. Dalal. Personalized social & real-time collaborative search. In *Proceedings of the International World Wide Web Conference (WWW '07)*, pp. 1285-1286, 2007.

[15] M. McPherson, L. Smith-Lovin, & J. Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology.* Vol. 27, No. 415, pp. 444-465, 2001.

[16] G. Salton & M. J. McGill. *Introduction to modern information retrieval*. McGraw-Hill, New York, 1983.

[17] Y. Hu, G. Xin, R. Song, G. Hu, S. Shi, Y. Cao, & H. Li. Title extraction from bodies of html documents and its application to web page retrieval. In *Proceedings of the ACM Conference on Information Retrieval (SIGIR* '05), pp. 250-257, 2005.

[18] S. Bao, G. Xue, X. Wu, Y. Yu, B. Fei & Z. Su. Optimizing web search using social annotations. In *Proceedings of the International World Wide Web Conference (WWW '07)*, pp. 501-510, 2007.

[19] T. H. Haveliwala. Topic-sensitive pagerank: A context-sensitive ranking algorithm for web search. *IEEE Transactions on Knowledge and Data Engineering.* Vol. 15, No. 4, pp.784-796, 2003.

[20] L. Pretto. A theoretical analysis of Google's PageRank. In *Lecture Notes in Computer Science.* Vol. 2476/2002 (Processing and Information Retrieval), pp. 125-136, 2002.

[21] T. Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining (SIGKDD'02)*, pp. 133-142, 2002.

[22] A. Hotho, R. Jäschke, C. Schmitz, & G. Stumme. Information retrieval in folksonomies: Search and ranking. In *Lecture Notes in Computer Science.* Vol. 4011/2006(the Semantic Web: Research and Applications), pp. 411-426, 2006.

[23] R. Schenkel, T. Crecelius, M. Kacimi, S. Michel, T. Neumann, J. X. Parreira, & G. Weikum. Efficient top-k querying over social-tagging networks. In *Proceedings of the ACM Conference on Information Retrieval (SIGIR '08)*, pp. 523-530, 2008.

[24] J. Haynes & I. Perisic. Mapping Search Relevance to Social Networks. In *Proceedings of the 3rd Workshop on Social Network Mining and Analysis (SNA-KDD)*, 2009.

[25] L. Gou, H. Chen, J. Kim, X. L. Zhang, & C. L. Giles. SNDocRank: A social network-based video search ranking framework. In *Proceedings of the ACM Conference on Multimedia Information Retrieval (MIR'10)*, pp. 367-376 2010.

[26] S. Wasserman & K. Faust. *Social network analysis: Methods and applications*. Cambridge Univ. Press, MA 1994.

[27] F. Lorrain & H. C. White. Structural equivalence of individuals in social networks. *Journal of Mathematical Sociology.* Vol. 1, pp. 49-80, 1971.

[28] E. A. Leicht, P. Holme & M. E. J. Newman. Vertex similarity in networks. *Physical Review E.* Vol. 73, No. 2, 026120, 2006.

[29] A. Clauset, M. E. J. Newman, & C. Moore. Finding community structure in very large networks. *Physical Review E.* Vol. 70, No. 6, 066111, 2004.

[30] M. E. J. Newman & M. Girvan. Finding and evaluating community structure in networks. *Physical Review E.* Vol. 69, No. 2, 026113, 2004.

[31] D. Coppersmith & S. Winograd. Matrix multiplication via arithmetic progressions. *Journal of Symbolic Computation*. Vol. 9, No. 3, pp. 251–280, 1990.

[32] L. Gou, J. Kim, H. Chen, J. Collins, M. Goodman, X. L. Zhang & C. L. Giles. MobiSNA: A mobile video social network application. In *Proceedings of the International Workshop on Data Engineering for Wireless and Mobile Access (MobiDE'09),* pp. 53-56, 2009.

[33] A. L. Barabasi, & E. Bonabeau. Scale-free networks. *Scientific American.* Vol. 288, No. 5, pp. 50-59, 2003.

[34] R. Landis & G. Koch. The measurement of observer agreement for categorical data. *Biometrics*. Vol. 33, pp. 159-174, 1977.

[35] K. Jarvelin & J. Kekalainen. IR evaluation methods for retrieving highly relevant documents. In *Proceedings of the ACM Conference on Information Retrieval (SIGIR '00)*, pp. 41-48, 2000.

[36] A. I. Schein, A. Popescul, L. H. Ungar & D. M. Pennock. Methods and metrics for cold-sart recommendations. In *Proceedings of the ACM Conference on Information Retrieval (SIGIR '02)*, pp. 253–260, 2002.