SNDocRank: a Social Network-Based Video Search Ranking Framework

Liang (Leon) Gou[1], Hung-Hsuan Chen[2], Jung-Hyun Kim[2], Xiaolong (Luke) Zhang[1], 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, hhchen, jzk171}@psu.edu, {lzhang, giles}@ist.psu.edu

ABSTRACT
Multimedia ranking algorithms are usually user-neutral and measure the importance and relevance of documents by only using the visual contents and meta-data. However, users' interests and preferences are often diverse, and may demand different results even with the same queries. How can we integrate user interests in ranking algorithms to improve search results? Here, we introduce Social Network Document Rank (SNDocRank), a new ranking framework that considers a searcher's social network, and apply it to video search. SNDocRank integrates traditional tf-idf ranking with our Multi-level Actor Similarity (MAS) algorithm, which measures the similarity between social networks of a searcher and document owners. Results from our evaluation study with a social network and video data from YouTube show that SNDocRank offers search results more relevant to user's interests than other traditional ranking methods.

Categories and Subject Descriptors
H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *relevance feedbacks, retrieval models, selection process.*

General Terms
Algorithms, Experimentation, Human Factors.

Keywords
Social Ranking, Multimedia Information Retrieval, Multilevel Actor Similarity.

1. INTRODUCTION
With the rapid growth of videos, photos, and audio media shared on the Internet, multimedia retrieval has become increasingly important. At the same time, users are increasingly engaged in social networking services like Facebook, Flickr, YouTube, etc. to communicate with their friends, family and colleagues by sharing videos, photos and other media. When social networks become accessible and large, we argue that effective multimedia information retrieval based on personal social contexts in social networks will be needed.

Currently, multimedia information retrieval is still largely based on integration of visual content and textual information [3, 5, 7, 12, 21, 25]. These approaches index and rank content with low-level visual features and high-level meta-data of multimedia content such as title, description, tags, category, author, etc. With the advent of media-sharing sites, such as Flickr and YouTube, the textual information used in retrieval can be extended to rich social contextual cues, such as social tags, geographical tags, and time and events contributed by communities [2, 13, 18, 28, 29]. These rich social context cues offer more meaningful information about the content of multimedia and may bridge the gap between low-level visual content and higher-level semantic concepts.

However, high-level social contextual cues, such as the interaction information embedded in a social network, have not been considered in multimedia information retrieval. Current ranking algorithms, for example, only focus on the content of multimedia and ignore the diverse needs of individual searchers. Usually, algorithms apply a global rank on all documents [7, 12, 21, 25] with the assumption that different users with the same query have the same information need. In reality, users belong to different social communities, and their social networks may implicitly include clues about their search needs and interests. For example, if a user wants to find a video about a friend, whose name happens to be the same as that of a celebrity, it is very likely that the search results returned will be more about the celebrity rather than the friend. However, if videos are searched within the user's social networks, the friend's information, rather than the celebrity's, will most likely come out first. An interesting problem then becomes how to use social network information to improve the performance of multimedia retrieval.

Here, we propose a framework to rank results based on a searcher's social contexts. We call the framework SNDocRank, which considers the features of the searcher's social network in the ranking of the relevance of documents. The premise of our methodology is that "birds of a feather flock together [17]": 1) users tend to be friends if they have common interests, and 2) users are more interested in their friends' information than other's information. We also propose a Multi-level Actor Similarity (MAS) measure, which is integrated in the SNDocRank framework to efficiently calculate a user similarity in large social networks. The results of our experiments on YouTube video search show that the SNDocRank method is more likely to return the documents that meet users' interests.

The paper is organized as follows. Section 2 reviews related works. In Section 3, we present the SNDocRank framework and the details of the MAS algorithm. Section 4 describes the experimental design and results. Finally, we conclude the paper with future research directions in Section 5.

MIR'10, March 29–31, 2010, Philadelphia, Pennsylvania, USA.

2. RELATED WORK AND BACKGROUND

Here we discuss relevant research on leveraging social factors to facilitate multimedia retrieval. We also introduce the concept of actor similarity in social network as a background for our framework.

2.1 Text-based Multimodal Multimedia Retrieval

In multimodal multimedia retrieval, to improve search performance, textual information such as associated captions, descriptions, and tags, is usually integrated into a multimodal approach with other channels of information, such as visual features and audio features. Among these different modalities, textual information largely alleviates the *semantic gap* [24] between low-level content and higher-level concepts to facilitate the ranking of searching results [7, 12, 21, 25] and to achieve better results than those obtained only by visual features [3].

Most studies integrate two separated modalities for visual content and texts, and then aggregate them to rank the final results. Some approaches only employ a single text-based, independent retrieval model [5] to re-rank the results. Others use multiple heterogeneous models by including different sources of text with different weighting strategies [7], or weighting visual contents over document-level context graph [10]. Other research applied pseudo-relevance feedback (PRFB) of text search results to video or image content [3, 12, 27]. The multimedia documents are labeled to be either pseudo-positive or pseudo-negative based on text search, and the labels are incorporated into the final ranking results.

However, these multimodal approaches rank all document with a global criterion without considering the individual context of a user's information needs. They also largely focus on document-level context information of visual content and do not consider the modalities dealing with higher level contextual information of multimedia, like social relationships of the owners of the multimedia document within different communities.

2.2 Community-contributed Multimedia Retrieval

Recently, there has been an increasing sharing and tagging of multimedia content on the web services like Flicker, FaceBook, and YouTube. The involvement of social actors has motivated interest in innovative approaches in multimedia retrieval tools that leverage community-contributed media collections.

Community-contributed multimedia retrieval approaches improve organizing and searching multimedia information based on various social cues such as social tags or annotations [2], geographical tags [13], and user-generated events [29]. The concurrency of social tags or annotations on different visual content indicates the semantic relations of visual content. Integrating the feature of visual content and social annotations can help improve any clustering problem with visual content and can alleviate any semantic gaps [2]. Geotag generated by users can be used to identify landmarks in visual content. SpiritTagger [18], for example, utilizes the GPS coordinates of photos and image contents to annotate photos with other geographically relevant tags. Some studies adopt a hybrid approach of two or more social cues to enhance the access to multimedia, e.g. a location-tag-vision-based approach to retrieving images of geography-related landmarks and features from the Flickr dataset [13]. ContextSeer [28] integrates the visual content, high-level concept scores, time and location metadata to improve search quality and recommend supplementary information, etc.

In summary, while many social cues in community-contributed multimedia retrieval, such as social tags, locations, and events, have improved the access to multimedia, those cues still largely focus on textual information and ignore the social contextual information embedded in social networks. The social cues of multimedia in communities indicate latent semantic relationships among multimedia, which may lead to better search results. Therefore, new ranking approaches incorporating community-level social network information are needed.

2.3 Actor Similarity in Social Networks

The actor similarity of social networks refers to how similar two actors in a social network are based on the structural information of the social network [26]. For example, in a teacher-student social network, if two teachers teach the same students of a class, the two teachers have connections with the same body of students. Thus, we can say the two teachers are similar in term of their social network with the students.

2.3.1 Cosine Actor Similarity

One common similarity measurement in social works is cosine similarity [26]. It is based on the idea of structural equivalence [16], which regards two actors similar if they share many neighbors in a social network. Cosine similarity measures the number of shared neighbors in a normalized way.

The cosine similarity S_{ij} between two vectors, i and j, is given by:

$$S_{ij} = \cos(i, j) = \frac{r_i \cdot r_j}{\|r_i\|_2 \cdot \|r_j\|_2} \quad (1)$$

Here denotes the dot-product of the two vectors, and r_i is a vector indicating the occurrence of other actor as neighbors of actor i.

However, cosine similarity only considers the shared neighbors that are directly connected to the two actors of interest. It ignores the global information of social networks.

2.3.2 LHN Vertex Similarity

LHN vertex similarity [15] expands the cosine similarity by going beyond direct neighbors. . The idea is that two vertices are similar if their immediate neighbors in a social network are themselves similar. Specifically, as shown in Figure 6, vertex i and v are connected (solid line), but v and j, i and j are not (dashed line). Then, how the vertex i is similar to vertex j is dependent on how the neighbor v is similar to j in a network. Obviously, this is a recursive concept, because the similarity between vertex v and j is related to the similarity between the neighbors of v and j. A start point this idea is that all vertices are similar to themselves.

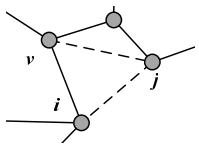


Figure 1. The diagram of vertex similarity.

One difficulty with LHN vertex similarity is the scalability. The computational complex of this approach is extremely high, because it involves expensive matrix multiplication. The LHN algorithm is not practical in real world applications, in particular when a social network involves millions of nodes and edges.

3. SNDocRank FRAMEWORK

In this paper, we propose a framework, SNDocRank, to rank the relevant documents by leveraging the actor similarity of a searcher and other users in a social network. In this section, the framework is first introduced, and then a new similarity algorithm, multi-level actor similarity (MAS), which is the core component in the SNDocRank framework, is presented to reduce the time complexity in the actor similarity calculation. Finally, we describe the integration of the MAS algorithm into the framework.

3.1 Framework of SNDocRank

The SNDocRank framework is shown in Figure 2. The framework is a central part in our search engine. In this system, a crawler collects two types of data: document meta-data, such as document titles, descriptions, tags, categories, and social network data, including users and relations among them.

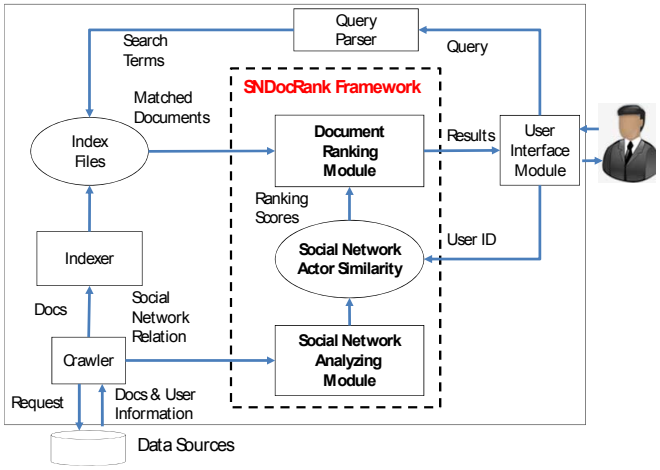


Figure 2. SNDocRank Framework.

We have three major modules in the SNDocRank framework: social network analyzing module, actor similarity module, and the document ranking module. The social network analyzing module receives, parses, and prepares social network data for actor similarity module. The actor similarity calculates similarity values among actors in social networks. We use our MAS method in this module, but as will shown later, other similarity algorithms, such as cosine similarity, can also be applied. In the document ranking module, actor similarity values are combined with document ranking values to deliver the final SNDocRank scores.

3.2 Multi-Level Actor Similarity (MAS)

The MAS method is proposed with two basic principles: first, it should consider the global structure information of a social network to enhance the accuracy of actor similarity measurement in the social network; second, it should reduce the complexity of similarity computation, and make the SNDocRank approach a feasible framework for various applications.

3.2.1 The MAS Approach

The MAS approach is based on the structural features of a social network, i.e. how actors are connected with each other in a social network. This approach involves three general steps. First, it clusters a social network hierarchically by using the network structure, and aggregates the clusters and edges among them. Then it applies a weighted LHN vertex similarity to the clustered networks at each level. Finally, global similarity values are calculated crossing all levels.

A social network is first clustered hierarchically, and at a specific level, each cluster can be regarded as a single abstract node. Thus, the network of abstract nodes at each level captures the main structural features of the network and can be treated as the backbone of the network at that level. In this way, the similarity between two clusters (abstract nodes) in the backbone network offers contextual information for the similarity between nodes within a cluster. When we calculate the similarity of any two nodes, there are two cases: if the two nodes belong to the same cluster, the similarity is only calculated within the cluster; if the two nodes belong to two different clusters, the similarity of two clusters as well as the similarity of the node and parent clusters are computed first and then combine two parts together into a final similarity value.

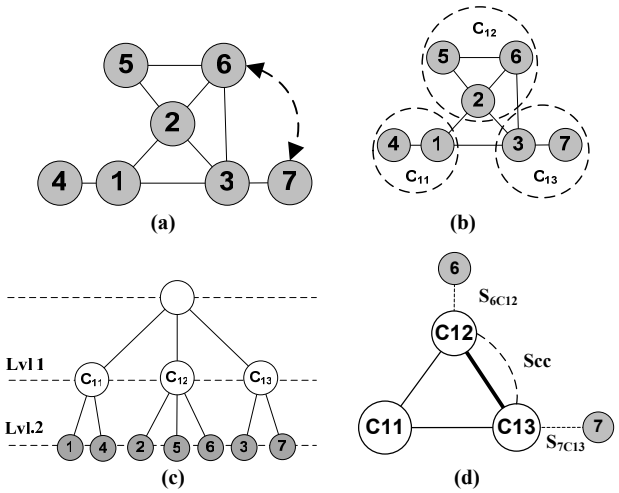


Figure 3. An example of Multi-level Actor Similarity (MAS).

Figure 3 illustrates the concept of the MAS algorithm. Suppose we have a social network in Figure 3a, and are going to calculate the similarity between Nodes 6 and 7. First, this network can be clustered hierarchically, shown in Figure 3b and 3c. In Figure 3b, Nodes 1, and 4 are grouped into one cluster—an abstract node C_{11}, Nodes 2, 5 and 6 into C_{12}, and Nodes 3 and 7 into C_{13}. After clustering, we have a backbone network consisting of C_{11}, C_{12}, and C_{13}, and the edges among them. The hierarchical structure is shown in Figure 3c. To calculate the similarity between Nodes 6 and 7, which belong to C_{12} and C_{13} respectively, we first computed the similarity values between C_{12} and C_{13} (S_{CC}), between Node 6 and Cluster C_{12} (S_{6C12}), as well as between Node 7 and Cluster C_{13} (S_{7C13}). Then we combine three similarity values together and get the final similarity value S_{6C12} S_{CC} S_{7C13}. Instead of computing the whole network in Figure 3a, we only consider the backbone network shown in Figure 3d. This approach reduces computation complexity without scarifying the global structural information of the social network.

3.2.2 Multi-Level Social Networks

3.2.2.1 Basic Notion in Social Networks

Some notations used in a social network are first defined in this section. A social network G can be represented as a set of actors (vertices or nodes), V, and relations (edges), E:

$$G = \{V, E\} \tag{2}$$

where V and E are sets of nodes and edges, respectively.

An adjacent matrix, A, represents the structure information of the social network G. A_{ij}, an element in adjacent matrix A, is defined as:

$$A_{ij} = \begin{cases} 1, \text{ if there is an edge between } i \text{ and } j, \\ 0, \text{ otherwise,} \end{cases} \tag{3}$$

In a weighted adjacent matrix WA, element WA_{ij} is the weighted value between two actors, i and j.

Thus, the relations of an actor, i, in a social network can be written as an actor vector:

$$r_i = \{A_{i1}, A_{i2}, ..., A_{in}\} \tag{4}$$

where n is the number of actors.

With the measurement of similarity, we can get a similarity matrix, S of a social network. Each element S_{ij} is the similarity value between two actors, i and j.

3.2.2.2 Hierarchical Clustering

The first step in MAS is to cluster social networks hierarchically. Social network clustering, also called community detection, is a continued topic of research in social networks. We use a fast community detection algorithm proposed by Clauset, et al. [4].

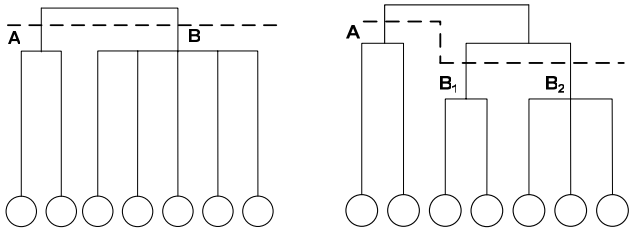
This algorithm is based on a quality measurement for clustering a network: ***modality*** [19]. A high value of modality indicates a good clustering of a network, which maximizes the number of edges within clusters and minimizes the number of edges between clusters. Modality is defined as [4]:

$$Q = \frac{1}{2m} \sum_{ij} \left[A_{ij} - \frac{k_i k_j}{2m} \right] \delta(c_i, c_j) \tag{5}$$

where A is an adjacent matrix, m is the number of edge in the network, k_i is the degree of node i, c_i is the cluster node i belongs to, and $\delta(\cdot)$ is the Kronecker function.

This fast hierarchical clustering method adopts an amalgamation strategy. The algorithm starts with all vertices as isolate clusters and follows a greedy approach. At each step, it tries to join all possible two clusters, calculates the increase of modularity ΔQ, and merges the two clusters with the greatest ΔQ into one cluster. This process is repeated until $\Delta Q \leq 0$ by joining any two clusters.

The result of the process is a hierarchical tree or dendrogram of network, shown in Figure 4. A cross-line on the tree at any level, as represented by a dotted line in Figure 4, gives the clusters at that level. The modality measurement offers us a criterion to decide where the cross-line should be place on the tree.



(a) unbalanced clustering **(b) balanced clustering.**

Figure 4. Two hierarchical clustering results.

However, one potential problem with this method is that the optimal clustered network is unbalanced. The size of some clusters may be very large, and some are very small. In Figure 4a, for example, Cluster A is small, but Cluster B is large. Our approach to address this issue is to apply this algorithm to the sub-clusters with size over a threshold, $N_k = n^{\frac{1}{2^d}}$, where n is the number of the whole network and d is the depth of sub-cluster. This produces a more balanced hierarchical tree (Figure 4b).

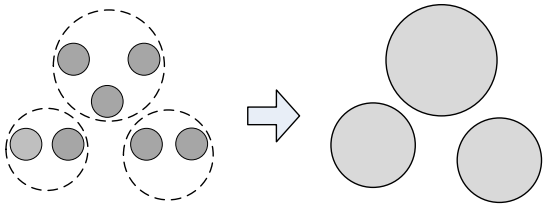
3.2.2.3 Aggregating Clustered Networks

To capture the main structural features of a clustered network at each level, we need to aggregate the clustered networks into abstract representatives of network. With aggregation, a cluster can be shown as one representative node, shown in Figure 5a, and all edges between node clusters can be treated as one single meaningful connection, shown in Figure 5b.

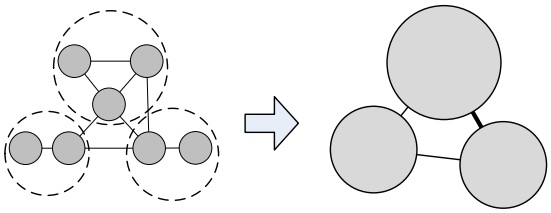
In this paper, we use the node with highest degree in the cluster to represent the cluster node: $C = \{i \mid max_{i \in C}(k_i)\}$, where k_i is the degree of node i. The aggregated edge is the sum of the number of edges between two clusters and indicates the connection strength of two clusters. This metric between two clusters u and w can be written as:

$$E(u, w) = \sum_{c_i = u, c_j = w} e(i, j) \tag{6}$$

where $e(i, j)$ is the edge between node i and j, and $c_i = u$, $c_j = w$ are mapping functions, indicating the nodes in u and w.



(a) node aggregation



(b) edge aggregation

Figure 5. Aggregating clustered networks.

3.2.3 Multi-Level Actor Similarity (MAS) Algorithm

3.2.3.1 Weighted LHN Vertex Similarity

We need to extend LHN vertex similarity to a weighted network. In the LHN vertex similarity algorithm, the adjacent matrix of the network is not weighted, and only contains 1 and 0 values. However, after clustering and aggregating the network, the adjacent matrices of clustered networks become weighted. The aggregated edge value between two clusters is the sum of the connections of the nodes within the two clusters.

A simple way to apply the LHN vertex similarity to a weighted adjacent matrix is to inverse the aggregated edge values of the weighted adjacent matrix and then to apply the LHN vertex similarity algorithm. The value of edge, 1 and 0, in an adjacent matrix of original settings of the LHN vertex similarity can be regarded as the reachability of two vertices in the network. Thus, if a weighted value of edge indicates the distance between two vertices, the LHN vertex similarity can be applied to the weighted adjacent matrix. The aggregated value of edge between two clusters indicates the strength of the clusters, and therefore, the inverse of this value can be treated as the distance between them. Then, we can obtain similarity matrix S by interactively computing:

$$DSD = \frac{\alpha}{\lambda_1} R(WA)(DSD) + I \qquad (7)$$

where $R(WA)$ is a matrix with the inverse of element in the weight adjacent matrix WA, D is the diagonal matrix with the degrees of the vertices in its diagonal elements, $D_{ij}=k_i\delta_{ij}$, A is an adjacent matrix, λ_1 is the largest eigenvalue of A, and α is a constant. The DSD can be easily computed iteratively with an initial value 0. This formula converges quickly and good convergence can be reached by 100 iterations and less [15].

3.2.3.2 Actor Similarity in A Multi-Level Network

We have three general scenarios when computing MAS, shown in Figure 6.

1) Two nodes belong to the same cluster. Similarity between the two nodes can be obtained by applying LHN to the sub-network within the cluster. For example, in Figure 6a, nodes 1 and 2 are within cluster C_{11}, and the similarity between them is computed within C_{11} using the LHN algorithm.
2) Two nodes are in two different clusters with the same level, as shown in Figure 6b. In this case, first we need to find the similarity between nodes and their direct parent clusters: Node 4 and Cluster C_{21}, and Node 5 and Cluster C_{22}. Then, the similarity between two clusters, Clusters C_{21} and C_{22} at the same level, is calculated within their parent node C_{12}. The final similarity is the multiplication of these three similarities.
3) Two nodes are in two different clusters at different levels, as shown in Figure 6c. This is similar to the second scenario except that the similarity between a node, 2, and a cluster, C_{21}, is calculated at the same level.

In general, to calculate the MAS, three steps are required: identifying minimal sub-tree of two nodes, calculating child-parent similarity, and calculating similarity of two nodes (either leaf node or cluster node) at the top level of the sub-tree.

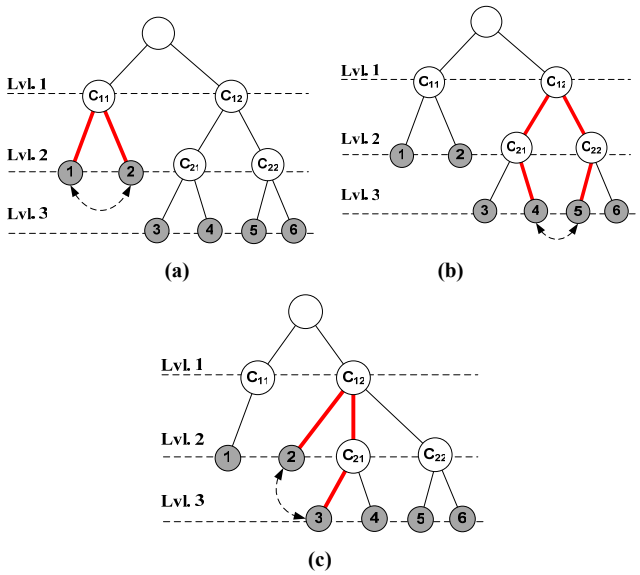


Figure 6. Three scenarios of MAS: (a) two nodes in the same cluster; (b) two nodes in two clusters at the same level; (c) two nodes in two clusters at different levels.

The first step is to find the lowest common ancestor of two nodes and obtain the sub-tree that the two nodes belong to. Thus, the similarity can be calculated within the sub-tree. Given two leaf nodes, i and j, we can search the lowest common ancestor, C_{hc} and the sub-tree, $T=\{V',E'\}$, where C_{hc} is the root of the sub-tree (h is the hierarchical level of the cluster C), V' is the set of nodes and clusters, E' is the parent-node connections. For example, the root cluster in Figure 6b is C_{11}.

The accumulated similarity of child-parent path of Node i can be written as:

$$S_{iC_{(h-1)i}} = S_{iC_{di}} S_{C_{di}C_{(d-1)i}} \cdots S_{C_{(d-m+1)i}C_{(d-m)i}} \qquad (8)$$

where $h\text{-}1=d\text{-}m$; C_{di} is the direct parent cluster of Node I; $C_{(h-1)i}$ is the ancestor cluster of Node i and a child of root cluster C_{hc}.

In the aggregation step, each cluster node is represented by a node with the highest degree within the cluster. The similarity between a node i (or cluster $C_{(d-m+1)i}$) and its parent cluster C_{di} (or cluster $C_{(d-m)i}$) is the similarity between the node i (or cluster $C_{(d-m+1)i}$) and the node with highest degree within the parent cluster, namely:

$$S_{iC_{di}} = \underset{C_{di}}{LHN}(i, C_{dj}), \text{ or} \qquad (9)$$

$$S_{C_{di}C_{(d-1)i}} = \underset{C_{(d-1)i}}{LHN}(C_{dj}, C_{(d-1)j}) \qquad (10)$$

Similarly, the accumulated similarity of child-parent path of node j is:

$$S_{jC_{(h-1)j}} = S_{jC_{dj}} S_{C_{dj}C_{(d-1)j}} \cdots S_{C_{(d-m+1)j}C_{(d-m)j}} \qquad (11)$$

The similarity of two nodes at the top level of the sub-tree is:

$$S_{C_{(h-1)i}C_{(h-1)j}} = \underset{C_{hc}}{LHN}(C_{(h-1)i}, C_{(h-1)j}) \qquad (12)$$

Therefore, the final similarity between Nodes i and j, S_{ij}, is the multiplication of three parts:

$$S_{i,j} = S_{iC_{(h-1)i}} S_{C_{(h-1)i}C_{(h-1)j}} S_{jC_{(h-1)j}} \quad (13)$$

3.2.4 Complexity Analysis

The complexity of MAS consists of two parts: hierarchical clustering and LHN algorithm at each level.

Let's first consider the hierarchical clustering. Suppose we have a social network with n nodes and m edges. The complexity of fast community detection is $O(ml\log n)$ [4], where l is the depth of the dendrogram describing the community structure. The fast community detection needs to be applied d times to generate hierarchical clustering, where d is the depth of the multi-level network. In a real-world network, we have $m \sim n$, and $l \sim \log n$, and $d \sim \log n$. Thus, the complexity of hierarchical clustering is

$$O(dm\log n) \sim O(n\log^3 n), \quad (14)$$

which is near linear time.

For the weighted LHN algorithm at multi-level, suppose the multi-level network is ideally well-balanced, the number of nodes in a cluster at depth i is $N_c = n^{\frac{1}{2^i}}$, and the number of clusters at depth i is $n / N_c = n^{1-1/2^i}$. In [15], the complexity of LHN vertex similarity is approximated by $O(n^{2.376})$ even with optimization by Coppersmith–Winograd algorithm [6]. Thus, the complexity of applying LHN algorithm at each level i is

$$O\left(\left(n^{1/2^i}\right)^{2.376} \cdot n^{1-1/2^i}\right) \quad (15)$$

By summing up the time complexity at each level, we have

$$O\left(\sum_{i=1}^{d} (n^{1/2^i})^{2.376} n^{1-1/2^i}\right) \sim O\left(dn^{1.688}\right) \sim O(n^{1.688}\log n) \quad (16)$$

,for $d \sim \log n$.

Now, let's considering both time complexity of hierarchical clustering and LHN for each level. Adding the complexity of two components together, the complexity of the whole algorithm is approximated by

$$O(n^{1.688}\log n + n\log^3 n) \sim O(n^{1.688}\log n), \quad (17)$$

because the complexity of hierarchical clustering, $O(n\log^3 n) < O(n^{1.688}\log n)$, can be dropped. This is much faster than original LHN, $O(n^{1.688}\log n) << O(n^{2.376})$.

3.3 Tf-idf

A basic ranking method is tf-idf [22] (term frequency-inverse document frequency). It is used to measure the importance of a particular term to a certain document based on the following observation: 1) a term is highly related to a document if the term appears many times in the document, measured by $tf_{i,j}$, and 2) a term is less important if it commonly appears in many different documents, measured by idf_i.

The tf-idf importance value of term t_i for document d_j is determined by multiplying $tf_{i,j}$ and idf_i.

$$tf\text{-}idf_{i,j} = tf_{i,j} \cdot idf_i \quad (18)$$

3.4 SNDocRank Score

SNDocRank score is the combination the basic tf-idf score and social network actor similarity value. SNDocRank first identifies the user who issues the queries, and ranks the search result based on the similarity scores with others in her social network. Thus SNDocRank score is given by:

$$SNDoc(v, t_i, d_j) = tf_{i,j} \cdot idf_i + \rho S_{vu} \quad (19)$$

where v is the current user, t_i is the term, d_j is the document, u the owner of the document, S_{vu} is the similarity value between user v and u in a social network, and ρ is a tuning parameter. The first term is the match score of queries and video meta-data, and the second part boosts the rank of the videos whose owners are similar with the user who fires the queries. S_{vu} could be any similarity score in social networks.

In this paper, we focus on using our MAS algorithm in SNDocRank framework. However, other similarity algorithms can also be used in the framework. This flexibility gives us the opportunity to compare not only SNDocRank with other ranking methods, but also different similarity methods within the SNDocRank.

4. EXPERIMENTS AND RESULTS

4.1 Simulated Social Network

In the first experiment, we compare MAS with cosine similarity and LHN similarity on a computer generated social network. The idea of the experiment is that we know the similarity of nodes in a simulated network, and then we test whether the similarity algorithms can reveal the node similarity based on the structure of the network.

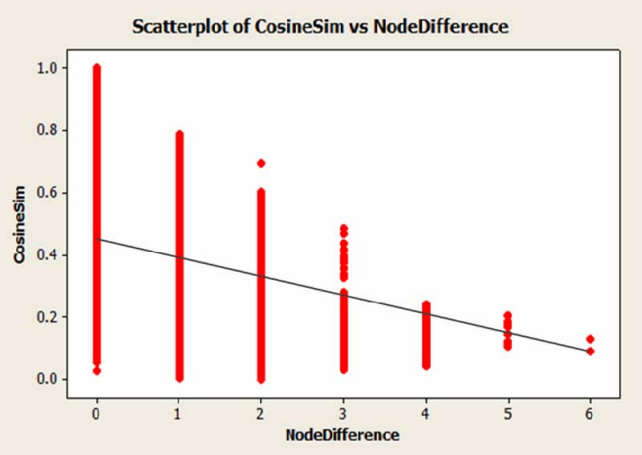
We generate a social network with node number n=2000, and each node is given a social property value which is random integer from 0 to 9. The edges were created between vertices with probability:

$$P(\Delta t) = p_0 e^{-a\Delta t} \quad (20)$$

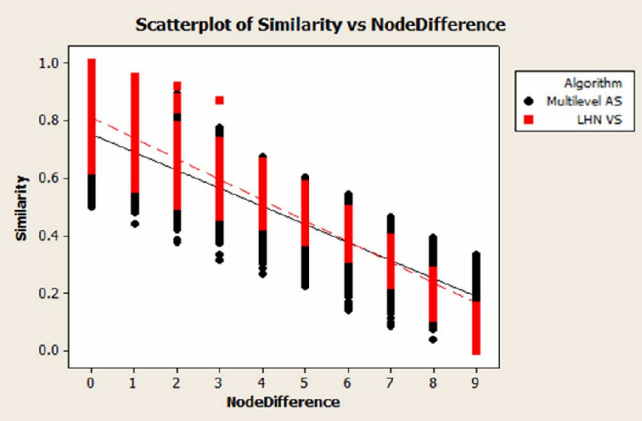
where Δt is the difference of the social property of two nodes which indicates how similar they are, α and p_0 are control parameters with values 3.0 and 0.2 in our experiment.

The experiment results are in shown in Figure 7. Figure 7 illustrates scatter (density) plots of the similarity values computed by three algorithms for all node pairs against the node differences in the model network. The similarity values of node pairs included in the plots are not directly connected by an edge, because the direct connected nodes have high similarity based on our assumption, and we are interested in not directly connected node pairs. The regress lines in the scatter plots are based on a least-squares fit. All similarity values are normalized.

The results in Figure 7 show that compared with cosine similarity algorithm, both MAS and LHN similarity is much more revealing measurement of actor similarity of in a network. The slopes of MAS and LHV are sharper than cosine similarity and the range of similarity values of MAS and LHN are narrower than cosine. The similarity values of MAS are comparable with those of LHN, shown in Figure 7. Although the slope of MAS is not as sharp as LHN, and the range of similarity is broader, MAS is still an effective algorithm in terms of complexity and accuracy.



(a)



(b)

Figure 7. Scatter plots of the computed similarities of all vertex pairs not directly connected against node differences: (a) cosine similarity; (b) Multilevel Actor Similarity vs. LHN similarity.

4.2 YouTube Dataset

In the second experiment, we implemented the SNDocRank framework in a mobile video social network application [9] and evaluated this framework and the MAS method with YouTube datasets. YouTube is an ideal case to demonstrate the usefulness of SNDocRank in that 1) each document (video) is associated with an owner who uploads the video, and 2) users have social networks of friendship.

We followed a breadth-first-search (BFS) strategy to retrieve users' social network and video information. Because YouTube has hundreds of thousands of registered users, we only randomly sample a very small portion compared to the total number of users. Specifically, we randomly selected a few users as target users and retrieved their information and video data. Next, we treated the target user's friends as new targets and retrieve the information of new targets. This procedure was performed recursively until no more targets were available or the number of retrieved users exceeded a pre-defined value.

4.2.1 Dataset Statistics

In the experiment, we retrieved 16,576 different registered users' information and 37,987 videos uploaded by these users from YouTube.

The retrieved social network has 42429 edges. The maximum degree is 89 and degree mean is 2.6. The degree distribution is shown in Figure 8 with log scales. It follows a power law distribution and has the scale-free feature of large social networks [1]. One interesting thing in this figure is that there is peak of frequency of degree with around 25, which can be used in our later experiment.

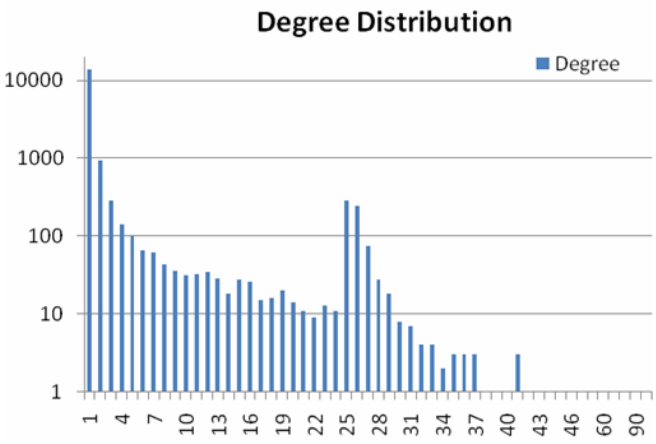


Figure 8. Degree distribution of YouTube social network

The betweenness distribution of retrieved social network is shown in Figure 9 with log scales. Most nodes have a very small betweenness value, less than 1E-6, and the nodes with betweenness value with from 5E-5 to 0.05 are some typical populations in the social network.

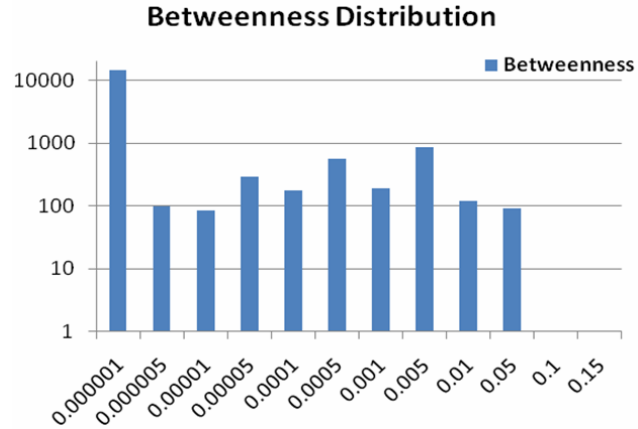


Figure 9. Betweenness distribution of YouTube social network

About the video data, there are 37987 videos and 1483 people who uploaded videos in the retrieved social network. The maximum number of videos upload by a person is 50. On average, the number of uploaded videos for each person is 25.62. The distribution of the number of users against the number of videos is shown in Figure 10.

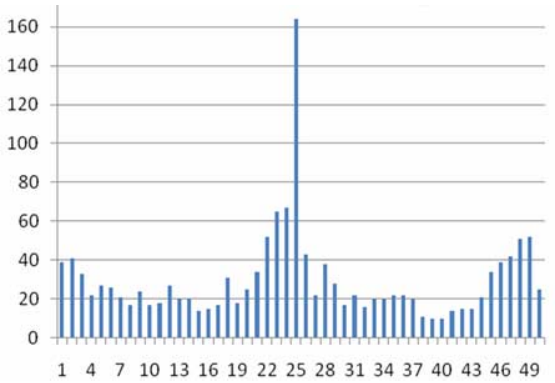


Figure 10. The distribution of the number of users (Y-Axis) against the number of videos they uploaded (X-Axis).

4.3 Evaluation Method

4.3.1 Interest Match Score (IMS)

The first evaluation is to match the returned results with both the interest of the user who fires queries and relevance of the results. The score is based on precision with matching the user's interest. The Interest Match Score (IMS) is defined by:

$$IMS = \frac{|\{\text{retrieved vids}\} \cap \{\text{relevant vids}\} \cap \{\text{interesting vids}\}|}{|\{\text{retrieved videos}\}|} \quad (21)$$

Matching the interest between the user and the returned results is based on the YouTube video categories, such as music, sports, and auto. If the category of a retrieved result is the same as the category of the user's interest, there will be one matching. The final score is normalized with the number of retrieved results.

Table 1 shows the 4 users and their profiles we selected from the retrieved YouTube social network in our experiment. The usernames in YouTube were replaced by anonymous id. Two users are interested in music and other two are interested in sports. In each interest category, the two users were selected with high degree and low degree respectively, which indicates their popularity in the social network. Three ranking algorithms were compared: tf-idf as the base line, SNDocRank with cosine similarity (Cos), and SNDocRank with MAS, We selected two ambiguous query terms for two categories of interest, "vampire" for the music users, and "cowboy" for the sport users.

Table 1. Users selected for experiment

UserId	Interest	Degree	Betweenness
888	Music	63	0.045190677
3883	Music	2	0.000120664
763	Sports	29	0.003873792
9656	Sports	1	0.0

4.3.2 Discounted Cumulative Gain (DCG)

In the second experiment with YouTube data, we use DCG to compare three ranking algorithms. DCG [11] measures the usefulness of the ranking result based on the relationship between the relevance scale of documents and the documents' positions in the ranking. The premise of DCG is that highly relevant documents are more useful when they have higher ranks in the result list. In our experiment, DCG is given by:

$$DCG_p = IRel_1 + \sum_{i=2}^{p} \frac{IRel_i}{\log i} \quad (22)$$

where $IRel_i$ indicates the level of relevancy and interest match for the result at position i. We have two levels of $IRel_i$.

$$IRel_i = \begin{cases} 1, \text{ if relevant and match the user's interest,} \\ 0, \text{ otherwise,} \end{cases} \quad (23)$$

In the experiment, 4 users with two categories of interests were used, shown in table 1. Twenty ambiguous terms were selected for each category of interest, shown in table 2. Video search results were ranked by three algorithms. Top twenty videos were mixed and presented to three PhD students to evaluate. The agreement among evaluators is examined by Kappa statistics [14].

Table 2. Queries used in evaluation.

Category	Queries
Music	candle, college, apple, road, spider, vampire, charismas, forest, friends, Michael, graduate, heart, bomb, ocean, rainbow, flower, angel, blue, rock, metal
Sports	bull, basketball, Pittsburgh, football, fan, boxing, field, defender, Jordan, highlight, kings, match, water, shoes, jump, slide, tackle, table, tiger, sock

4.4 Results

4.4.1 IMS

IMS, similar to precision, shows how relevant the returned videos are and whether the returned videos match the user's interests.

The IMS results of users with two categories of interest: music and sports are shown in Figure 11 and Figure 12. In each category of interest, two users and three algorithms are compared. The users can have high degree (popular/active users) and low degree (unpopular/active users) in their social networks. Three algorithms are examined.

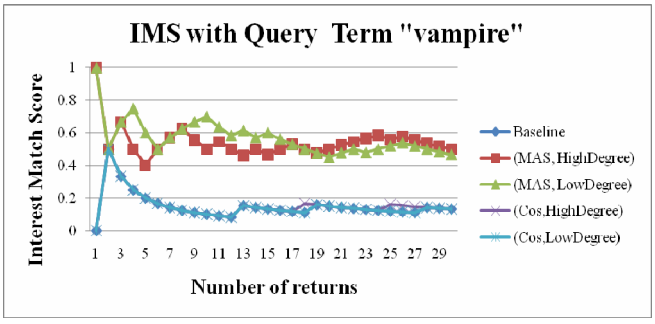


Figure 11. Interest match score for two users with interest of music but different degrees for term "vampire".

Figure 11 shows that the IMS for two users with interest of music and term "vampire" with 30 returns. As shown, the IMS scores returned by the MAS method are much higher than those returned by baseline and cosine similarity, no matter whether the user's degree is high or low. The IMS scores of cosine approach with high degree are slightly better than those of base line and cosine

with low degree. In terms of MAS, there is little difference between the IMS of high degree and low degree users.

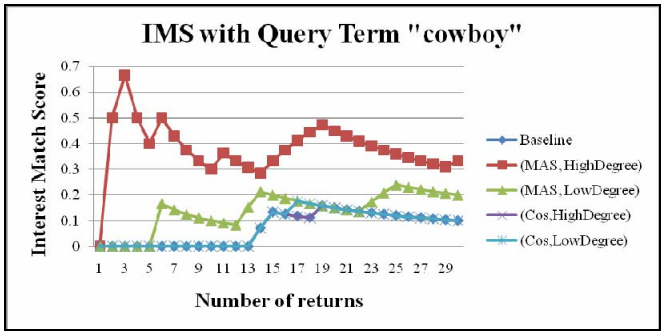


Figure 12. Interest match score for two users with interest of sport and different degrees and term "cowboy".

Figure 12 shows that the IMS for two users with interest of music and term "cowboy" with 30 returns. It shows that the results returned by MAS and with the high degree user have the highest IMS alongh with all returns. The IMS of the cosine approach is still close to the baseline. One interesting results is that the performance of MAS with low degeree user is much lower than MAS with high degree user, compared with the reults shown in Figure 12. It indicates that the interest of users have influence on the performace of MAS.

In summary, the IMS of the MAS method are higher than those of tf-idf baseline and social network cosine similarity with different interests and degrees of users. Besides, within MAS, users' interets exert an influence over the IMS values.

4.4.2 DCG

While IMS examines the relevancy and matching of interests of returned results, DCG evaluates the performance of ranking approaches. The average DCG of users with two categories of interest: music and sports are shown in Figure 13 and Figure 14. Similar with the results in section 4.4.1, in each category of interest, two users and three algorithms are compared.

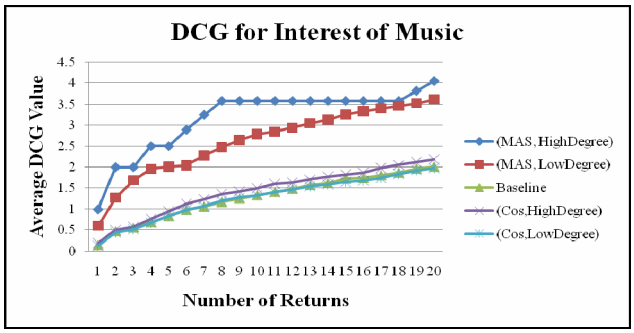


Figure 13. Average DCG for two users with different degrees and interest of music

Figure 13 shows the average DCG for two users with different degrees and interest of music. Obvisously, the average DCG values of MAS are higher than those of tf-idf and cosine along with all numer of returns. Cosine approaches have the similar average DCG value with tf-idf. Both MAS and cosine apporaches perform better for high-degree users than low-degree users.

In Figure 14, the interests of users are changed to sports. We can also get the same conclusion that the performace of MAS outperforms baseline and cosine. Supprisingly, in terms of the cosine appraoch, the user with low degree performs slightly better than the one with high degree.

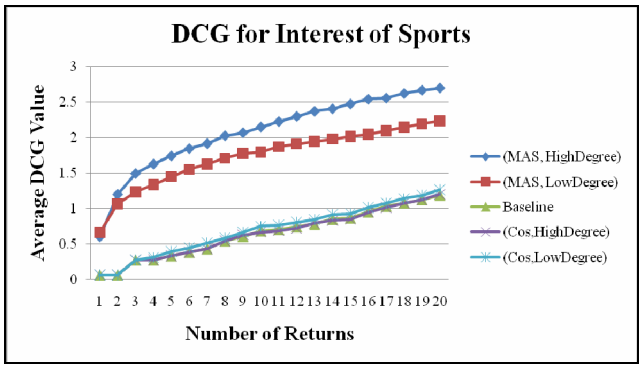


Figure 14. Average DCG for two users with different degrees and interest of sports.

In summary, the MAS outperforms than the tf-idf baseline and social network cosine similarity in terms of the effectiveness of the ranking results. The advantages of MAS are consistent with users of different interests and degrees.

4.5 Discussion

Overall, with SNDocRank framework, the MAS performances better than basic tf-idf and cosine similarity in terms of the relevancy and matching of interests of returned results with searchers, and the ranking effectiveness. The better performance of MAS keeps consistent with searchers of different sizes of social networks, interests, and degrees. This indicates that the structure of users' social network can provide some clues about the users' information needs.

However, there are some interesting observations that may offer us some insights into the performance of MAS in the experiments.

The degree of user in social networks has influences over the performance of MAS and cosine similarity in the SNDocRank framework. Generally speaking, both MAS and cosine of the user with high degree performance better than those of the user with low degree. This indicates that if a user wants to get more relevant and interesting results, the user should make more friends who share the similar interests. With a higher degree, the user can disseminate the information about her preference more widely in the social network.

On the other hand, MAS is less sensitive than cosine approach in terms of degree. MAS with low degree user still works better than cosine approach with high degree, because MAS considers the global information of social networks, but cosine approach only focuses on the direct neighbors.

Bias of the categories of videos and users' interests also affects the performance of MAS. The differences of IMS of MAS between high degree and low degree with interest of sports, shown in Figure 12, are larger than those with interest of music, shown in Figure 11. It is probable that there are more users shared interest of music than those who are interested in sports and the categories of videos are mainly about music in YouTube. The observation that the total percentage of users whose interest is sports is small in the retrieved social network verifies our

hypothesis. Thus, MAS has difficulty in improving the ranking of the biased information. One potential solution is to detect the communities of interest in social networks, and rank the videos based on the interest communities.

5. CONCLUSION AND FUTURE WORK

In this paper, we present a SNDocRank framework, which incorporates both the videos meta-data and the social networks of owners to rank results. With the assumption that "bird of a feather flock together", the SNDocRank framework ranks the videos based on the similarity of the owners of videos in social networks. To tackle the problem of actor similarity computation in large social networks, we propose a multilevel actor similarity (MAS), which is plugged into the SNDocRank framework. The experiment of a simulated network shows the effectiveness of MAS. With YouTube dataset, the MAS method outperforms the cosine similarity in the SNDocRank framework in terms of the relevancy, matching of interests of returned results, and the ranking effectiveness. Some implications of the SNDocRank approach are also drawn based on the experimental results.

For the future work, we are interested in extending this work in two directions. First, we will incorporate some visual content detection into our framework to improve the multimedia retrieval performance. On the other hand, we will apply our approach to other large social networks to examine its performance.

6. ACKNOWLEDGEMENTS:

Part of this work has been funded by Alcatel-Lucent. We also acknowledge useful discussions with Marc Goodman and Jason Collins.

7. REFERENCES

[1] Barabasi, A. L., & Bonabeau, E. Scale-free Free Networks. *Scientific American*, 288(5), pp.50-59, 2003.

[2] Bellucci, A., Ghiron, S. L., Aedo, I., & Malizia, A. Visual tag authoring: picture extraction via localized, collaborative tagging. In *ACM AVI*, pp. 351-354, Napoli, Italy, 2008.

[3] Chang, S.-F. et al, Columbia University TRECVID-2006video search and high-level feature extraction, *NIST TRECVID workshop*, 2006.

[4] Clauset, A., Newman, M. E. J., & Moore, C. Finding community structure in very large networks. *Phy. Rev. E*, 70(6), 66111, 2004.

[5] Clough, P., Grubinger, M., Deselaers, T., Hanbury, A., & Muller, H. Overview of the imageclef 2007 photographic retrieval task. In *CLEF 2007*, vol. 5152 of *LNCS. CLEF*, Springer-Verlag, 2008.

[6] Coppersmith, D. & Winograd, S. Matrix multiplication via arithmetic progressions. *Journal of Symbolic Computation*, 9:251–280, 1990.

[7] Escalante, H. J., Hérnadez, C. A., Sucar, L. E., & Montes, M. Late fusion of heterogeneous methods for multimedia image retrieval. In *ACM MIR*, pp. 172-179, Canada, 2008.

[8] Goodrum, A. Image information retrieval: An overview of current research. *Journal of Informing Science*, 3(2), 2000.

[9] Gou, L., J. Kim, H. Chen, J. Collins, M. Goodman, X. L. Zhang & C. L. Giles. MobiSNA: a Mobile Video Social Network Application. *Proc. of MobiDE'09*: 53-56.

[10] Hsu, W. H., Kennedy, L. S., & Chang, S. Video search reranking through random walk over document-level context graph. In *ACM Multimedia*, pp. 172-179, Germany, 2007.

[11] Jarvelin, K. & Kekalainen, J. IR evaluation methods for retrieving highly relevant documents, *Proc. of the 23rd SIGIR conference*, pp.41-48, 2000.

[12] Kennedy, L. S. & Chang, S. A reranking approach for context-based concept fusion in video indexing and retrieval. In *ACM CIVR*, The Netherlands, pp. 333-340, 2007.

[13] Kennedy, L., Naaman, M., Ahern, S., Nair, R., & Rattenbury, T. How flickr helps us make sense of the world: context and content in community-contributed media collections. *ACM Multimedia*, pp.631-640, Germany, 2007.

[14] Landis, R. J. & Koch, G. The measurement of observer agreement for categorical data. *Biometrics,* 33, pp.159-174, 1979.

[15] Leicht, E. A., Holme, P., & Newman, M. E. J. Vertex similarity in networks. *APS*, Vol. 73, pp.26120, 2006.

[16] Lorrain, F. & White, H. Structural equivalence of individuals in social networks. *Math. Sociol*, 1, pp. 49-80, 1971.

[17] McPherson, M., Smith-Lovin, L., et al. Birds of a Feather: Homophily in Social Networks. *Annual Review of Sociology*, 27(415): 444, 2001.

[18] Moxley, E., Kleban, J., & Manjunath, B. S. Spirittagger: a geo-aware tag suggestion tool mined from flickr. In *ACM MIR*, pp.24-30, Canada, 2008.

[19] Newman, M. E. J. & Girvan, M. Finding and evaluating community structure in networks. *Phys. Rev. E*, 69, pp.26113, 2004.

[20] Rautiainen, M. M. & Seppdnen, T.. Comparison of visual features and fusion techniques in automatic detection of concepts from news video. In *Proc. of the IEEE ICME*, pp.932-935, 2005.

[21] Rautiainen, M., Ojala, T., & Tapio, S. Analysing the performance of visual, concept and text features in content-based video retrieval. In *ACM MIR*, pp.197-204, New York, NY, USA, 2004.

[22] Salton, G. & Buckley, C. Term-weighting approaches in automatic text retrieval. *Info. Proc. & Mgt.*, 24 (5), pp.513–523, 1988.

[23] Salton, G. Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer. Addison-Wesley, Reading, MA, 1989.

[24] Smeulders, A. W.,Worring, M., Santini, S., Gupta, A., , & Jain, R. Content-based image retrieval at the end of the early years. *IEEE Trans. Pattern Anal. Mach. Intell.* 22(12), pp.1349–1380, 2000.

[25] Snoek, C., Worring, M., and Smeulders, A. Early versus late fusion in semantic video analysis. In *ACM Multimedia*, pp. 399-402, Singapore, 2005.

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

[27] Yan, R., Hauptmann, A., & Jin, R.. Multimedia search with pseudo-relevance feedback. In *AAAI Spring Symposium on IMKM*, Palo Alto, CA, 2003.

[28] Yang, Y. H., Wu, P. T., Lee, C. W., Lin, K. H., Hsu, W. H., & Chen, H. H. ContextSeer: context search and recommendation at query time for shared consumer photos. In *ACM Multimedia*, pp.199-208, Canada, 2008.

[29] Yuan, J., Luo, J., Kautz, H., & Wu, Y. Mining GPS traces and visual words for event classification. In *ACM MIR*, pp.2-9, Canada, 2008.