Tag Clouds Revisited

Dimitrios Skoutas
L3S Research Center
Hanover, Germany
skoutas@L3S.de

Mohammad Alrifai
L3S Research Center
Hanover, Germany
alrifai@L3S.de

ABSTRACT

Tagging has become a very common feature in Web 2.0 applications, providing a simple and effective way for users to freely annotate resources to facilitate their discovery and management. Subsequently, tag clouds have become popular as a summarized representation of a collection of tagged resources. A tag cloud is typically a visualization of the top-k most frequent tags in the underlying collection. In this paper, we revisit tag clouds, to examine whether frequency is the most suitable criterion for tag ranking. We propose alternative tag ranking strategies, based on methods for random walk on graphs, diversification, and rank aggregation. To enable the comparison of different tag selection and ranking methods, we propose a set of evaluation metrics that consider the use of tag clouds for search, navigation and recommendations. We apply these tag ranking methods and evaluation metrics to empirically compare alternative tag clouds in a dataset obtained from Flickr, comprising 488,112 tagged photos organized in 451 groups, and 112,514 distinct tags.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; H.5.1 [**Information Interfaces and Presentation**]: Multimedia Information Systems

General Terms

Algorithms, Measurement

Keywords

Web 2.0, Tagging, Web Resources

1. INTRODUCTION

With the abundance of content on the Web, especially with the rapidly increasing amount of user generated content in Web 2.0, retrieving and managing information remains a significant and challenging problem. Although content-based retrieval usually performs sufficiently well for text, such as documents or Web pages, the same does not hold for other types of content. For example, for multimedia resources or bookmarks, it has become clear that exploiting metadata is often crucial for achieving good performance. However, although the Semantic Web aims at annotating Web resources to facilitate their search and management, attempts in this direction are hindered by the significant effort required to create and maintain formal annotations, as well as the ontologies on which they rely upon. In contrast, *tagging* has emerged as a very simple, easy and effective solution for annotations. Most Web 2.0 applications allow users to tag resources created by themselves or others, and exploit these tags to improve search, navigation and recommendations [21].

The fact that tagging is such an easy and fast process that any user can perform has been the main driving factor for its widespread adoption by most Social Web applications, such as Flickr, Delicious, Technorati, Facebook or Last.fm. However, this also makes the results of the tagging process very noisy. To address this problem, several approaches have proposed methods for extracting semantics from tags or computing the relevance of a tag to a resource in order to improve or recommend the assignment of tags to resources (e.g. [14, 12, 18, 23, 13, 7]). These approaches focus on (re-)ranking or suggesting tags for an *individual* resource, based on its content (e.g., visual features of an image), as well as the content and tags of other similar resources.

In this paper, we consider instead the problem of selecting and ranking tags to describe *groups* of tagged resources. For example, in Flickr or Facebook, users can create albums to organize their photos; similarly, in Last.fm, they can create playlists containing tracks from similar artists or for specific occasions. Such item collections can become arbitrarily large over time. Relying on a title or a short text to describe the group contents is not sufficient and may easily become outdated or reflect only the owner's perspective. On the other hand, listing all the contents is impractical for large collections, while selecting only a few representatives to display may not be straightforward. When dealing with collections of tagged resources, tag clouds have become a common way for describing the group contents and allowing navigation.

The goal of a *tag cloud* is to display the most relevant and important tags for the items in a group. In practice, tag clouds typically display the most frequently occurring tags, since this is both intuitive and easy to compute. Figure 1

CIKM'11, October 24–28, 2011, Glasgow, Scotland, UK.

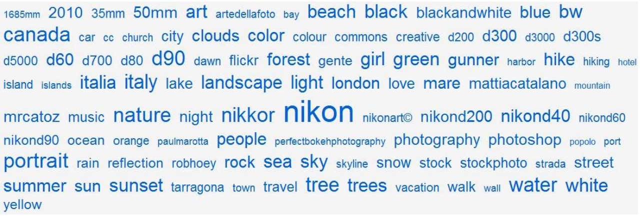

1685mm 2010 35mm 50mm art artedellafoto bay beach black blackandwhite blue bw
canada car cc church city clouds color colour commons creative d200 d300 d3000 d300s
d5000 d60 d700 d80 d90 dawn flickr forest gente girl green gunner harbor hike hiking hotel
island islands italia italy lake landscape light london love mare mattiacatalano mountain
mrcatoz music nature night nikkor nikon nikonart© nikond200 nikond40 nikond60
nikond90 ocean orange paulmarotta people perfectbokehphotography photography photoshop popolo port
portrait rain reflection robhoey rock sea sky skyline snow stock stockphoto strada street
summer sun sunset tarragona town travel tree trees vacation walk wall water white
yellow

Figure 1: Example of a tag cloud from Flickr.

shows an example of a tag cloud displayed by Flickr for a group of photos with title "This is nikon art".

In this paper, we deal with the problem of *selecting and ranking tags for tag clouds*, addressing two main questions: (a) how effective is the strategy of ranking tags in item collections based on their frequency, and (b) are there any better strategies for this task? To address these questions, we propose and examine alternative methods to select and rank tags in groups of tagged objects, and we compare these methods on a large real-world dataset containing groups of tagged photos obtained from Flickr. To compare the different methods, we measure their effectiveness in terms of a set of proposed metrics that characterize the usefulness of a tag cloud for search and navigation. Moreover, we measure their accuracy for the task of recommending groups for a tagged item, when groups are represented by different tag clouds. The results of our evaluation show that although ranking tags based on their frequency performs well in most cases, even more effective rankings can be obtained using other methods. Specifically, our contributions can be summarized as follows:

- We formalize the problem of ranking tags to represent groups of tagged objects, and we propose different ranking strategies that go beyond the simple frequency-based method.

- We propose a set of metrics to objectively and automatically evaluate the effectiveness of alternative tag clouds for a given group of objects.

- We empirically evaluate the presented tag selection and ranking methods in terms of the described evaluation methodology, using a large real-world dataset from Flickr, comprising 451 groups, 488,112 tagged photos, and 112,514 distinct tags.

The rest of the paper is organized as follows. The next section discusses related work. Section 3 introduces the basic framework and notation. In Section 4, we present different strategies for selecting and ranking tags for tag clouds. Section 5 proposes a set of evaluation metrics that allow to objectively compare different tag clouds for the tasks of search, navigation and recommendations.Section 6 presents the results of our experimental evaluation on a large collection of tagged photos and groups obtained from Flickr. Finally,Section 7 concludes the paper.

2. RELATED WORK

Due to the popularity and wide adoption of manual tagging of resources in content sharing Web sites, this area has attracted a lot of research interest. In the following, we review some of the main related research efforts, including also some related work on faceted browsing.

2.1 Tag Ranking

The most relevant work to the one presented in this paper is a recent effort described in [22], which addresses the problem of selecting tags to summarize query results. It combines frequency and diversity to increase the coverage of the query results. However, it does not consider other methods such as tag co-occurrence or rank aggregation or different variations for diversification as in this paper. In addition, it proposes a set of metrics for evaluating different tag clouds. Although some of these metrics are common to our work, such as coverage and overlap, it focuses on measures for relevance, balance and cohessiveness of results, while we are more interested in characteristics such as selectivity, navigation cost and accuracy for recommendations.

Some recent efforts have dealt with the problem of (re-) ranking the set of tags assigned to an object. [14] focuses specifically on images and computes a relevance score between an image and each tag assigned to it by performing a random walk on a graph representing tag similarities. Apart from tag co-occurrences, these similarities take into consideration the content of the photos using low-level visual features. In the same direction, [12] estimates the relevance of a tag to an image based on its occurrence or not in visually similar images. Furthermore, extracting event and place semantics from tags of photos has been investigated in [18].

In contrast, our goal is to select and rank tags for groups of objects rather than each single object. However, it is interesting to examine how the methods and metrics presented in our work can be adapted for individual resources or extended to exploit semantics and content-based similarity.

2.2 Tag Recommendations

To reduce the noise and the effort in tagging, several approaches have focused on recommending tags to users for a given resource. In [23], tag recommendation for images is formulated as a learning problem, proposing a multimodality recommendation based on both tag and visual correlation. [13] proposes a re-tagging scheme for images that maintains consistency of visual and semantic similarity between tags and images. In [2], tags are propagated along edges in a graph connecting similar documents. A graph-based ranking algorithm is proposed in [7] for personalized tag recommendation. When a user issues a tagging request, both the resource and the user are treated as queries, accounting for relevance and personalization, respectively. Then, the top ranked tags are recommended to the user.

Again, these approaches focus on recommending tags for individual objects and they rely on content-based (visual) similarity for relevance.

2.3 Visualization of Tag Clouds

In a different line of research, the impact of the visualization aspects of tag clouds on the user experience has been studied. A comparative study of several tag cloud layouts (e.g. sequential, circular and clustered layouts) has shown that they clearly affect task performance, and thus should be carefully designed [15]. Another experimental evaluation indicated that semantically clustered tag clouds can provide improvements over random layouts in specific search tasks, increasing also the attention towards tags in small fonts [20]. A method for displaying large-scale tag clouds using a topographical image has been proposed in [5].

Visualization and layout aspects are not considered in our work. However, it is interesting to examine how the proposed tag selection and ranking algorithms can be combined with different tag cloud layouts. For example, a clustered layout might be more suitable for a tag diversification algorithm, while a sequential layout for a rank aggregation method.

2.4 Faceted Browsing

Finally, our work has many commonalities with methods for automatically selecting facets and facet-value pairs in faceted browsing interfaces, which are also used in many applications to help users navigate through large collections of resources [8]. Facets are attributes describing resources in a collection. For example, typical facets for scientific publications are *author*, *title*, *conference*, *year*, etc. Each facet may have a very large list of possible values; however, only a few facets and facet-value pairs can be displayed in a user interface. Thus, several works have focused on the problem of ranking facets and facet-value pairs, going beyond the standard frequency-based approach.

Facetedpedia [11] automatically selects and ranks facets in wikipedia, based on a user navigation model to define the cost of different facets. This is extended with a definition of similarity between facets to allow the ranking of combinations of facets. To make the navigation model manageable, some simplifying assumptions are made; for example, deselecting a previously selected facet is not allowed. A set of metrics for determining which facets constitute good "navigators" in RDF datasets is proposed in [17]. These include *predicate balance*, which compares the sizes of the result sets of different facet-value pairs, *object cardinality*, which is the number of possible values for a facet, and *predicate frequency*, which is the number of objects that have a non-null value for a given facet. To select facets that minimize the navigation cost in a database, [19] finds the minimum cost decision tree that distinguishes each tuple from other tuples based on its attributes values. This is similar, to some extent, with our metric for selectivity. Finally, personalized and interactive faceted search is studied in [9], focusing on a methodology to evaluate alternative selections of facets. A user simulation model is used to define a utility score for alternative faceted search interfaces.

Summarizing, since tag clouds and faceted browsing interfaces serve similar purposes, there are some similar ideas underlying methods, models and metrics in both categories. However, due to the different nature of facets, which are structured and have the form of attribute-value pairs, those approaches are usually not suitable or directly applicable for tag selection and ranking.

3. TAG SELECTION FRAMEWORK

Assume a set of objects $\mathcal{U}$, a set of (possibly overlapping) groups $\mathcal{G}$, and a set of tags $\mathcal{T}$. Objects are assigned to groups according to a mapping function $m_g : \mathcal{U} \times \mathcal{G} \rightarrow \{0, 1\}$. Similarly, tags are assigned to objects according to a mapping function $m_t : \mathcal{U} \times \mathcal{T} \rightarrow \{0, 1\}$.

Let $T(u)$ denote the set of tags assigned to an object u. Similarly, $U(t)$ denotes the set of objects tagged with t. Typically, the latter refers to the objects within a particular group G under consideration, but to simplify notation we omit G since it is clear from the context.

Given a group G, the set $T(G)$ of all the tags related to it is the union of the sets of tags assigned to its objects, i.e.,

$$T(G) = \bigcup_{u \in G} T(u) \tag{1}$$

This set can be arbitrarily large, depending on the size of the group and the number and variety of tags assigned to its objects. Applications need to select and visualize a few of these tags to represent a summary of the group contents. Thus, the goal is to select a subset $T_G \subseteq T(G)$ of size k (for a given, relatively small, integer k) to describe the group. In addition, the tags in T_G should be ranked, since this ranking is typically taken into consideration when visualizing a tag cloud. Therefore, we consider T_G as an ordered list of tags $\{t_1, t_2, \ldots, t_k\}$. We denote the rank of a tag t in the tag cloud T_G by $r(t)$ (with the first tag having $r(t) = 0$).

Let $f(t)$ be a scoring function that assigns to each tag t of a tag cloud T_G a utility value in the interval $[0, 1]$. We define the overall utility value of T_G as

$$F(T_G) = \frac{\sum\limits_{t \in T_G} \alpha(r(t)) \cdot f(t)}{|T_G|} \tag{2}$$

where $\alpha()$ is a discount function that adjusts the utility of each tag according to its position in the cloud, e.g.:

$$\alpha(i) = \frac{1}{\beta \cdot i + 1} \tag{3}$$

where the parameter β determines the rate of reduction.

Given a group G and an integer k, the optimal tag cloud for G is the set T_G that is a subset of $T(G)$ with size k and maximizes the utility function F:

$$T_G = \underset{T \in 2^{T(G)} \& |T|=k}{\arg\max} F(T) \tag{4}$$

Therefore, the main question is how to specify the utility function f, and subsequently, to maximize it. In the following section, we propose different tag selection methods based on different approaches for defining the utility function f for the members of the tag cloud.

4. TAG SELECTION STRATEGIES

We consider different strategies for ranking tags to describe sets of tagged objects. In particular, we first start with the standard frequency-based ranking, and we examine two possible extensions. Then, we present two methods based on diversification algorithms, and finally we present a method based on an algorithm for rank aggregation.

4.1 Based on Frequency

4.1.1 *Frequency scoring*

The simplest, most straightforward and most widely used approach for selecting the contents of a tag cloud for a set of objects is to rank tags based on their frequency, i.e., the number of objects to which a tag is assigned. This is based on the assumption that, if a large number of objects in the group has been tagged with a tag t, then t has high utility for describing the contents of the group. Therefore, the utility function of a tag $t \in T(G)$ is defined in this case as

$$f(t) = \frac{|U(t)|}{|G|} labeleq : frequency \tag{5}$$

Then, the tag cloud is simply formed by selecting the top k most frequent tags. In practice, it is possible that some objects are more important or relevant for the group than others. Equation **??** can be easily extended to include a weight parameter for each object. For example, these weights could be determined based on the number of views for each item. For simplicity, we assume in this paper that all objects have equal weights.

In the following, we examine two other approaches for defining the utility function f, which can be seen as extensions of the frequency based approach. Since the frequency of a tag, as defined in Equation **??**, is used as basis for these and also for subsequent definitions of the utility function, we denote it as $fr(t, G)$ for later reference.

4.1.2 TF.IDF scoring

The first extension is to rank tags based on the same idea as $tf.idf$ scoring for document retrieval. This is based on the assumption that a tag has lower utility in describing the contents of a group if it also occurs frequently in several other groups. In other words, the computation of the utility score of a tag t with respect to a group G relies not only on the contents of this particular group but also on the contents of the other groups in the collection. In particular, we can define the utility function as

$$f(t) = fr(t, G) \cdot idf(t) \tag{6}$$

where the idf of a tag in the collection is computed by

$$idf(t) = \log \frac{|\mathcal{G}|}{|\{G \in \mathcal{G} \,:\, t \in T(G)\}|} \tag{7}$$

The tag cloud comprises the top k tags in terms of $f(t)$.

4.1.3 Graph-based scoring

The second extension is based on the assumption that co-occurrence of tags is important. Typically, tags assigned to objects are keywords freely chosen by the users without restricting them to the use of an ontology or controlled vocabulary. This significantly facilitates the process of tagging and has contributed to its popularity and widespread adoption, but as a consequence tags may often be ambiguous or used in different ways by different users. For this reason, considering combinations of tags that occur together rather than individual tags may be more informative, since it provides context information.

To take this into consideration when constructing the tag cloud of a group G, we create a graph of the tags in $T(G)$, where an edge between two tags t_i and t_j denotes that there is an object in G tagged with both t_i and t_j. Then, the utility score of a tag is derived by performing a random walk on this graph [16]. Note that this is similar to the method applied in [14] for ranking the tags of individual objects.

Initially, the score of each tag t is set to its frequency, i.e., $f_0(t) = fr(t, G)$. The transition probability from a tag t_i to a tag t_j is computed as

$$p(t_i, t_j) = \frac{sim(t_i, t_j)}{\sum\limits_{t \in T(G)} sim(t_i, t)} \tag{8}$$

The similarity $sim(t_i, t_j)$ between two tags can be computed using the Google similarity distance [3], which takes into account the number of objects tagged with each tag, the number of objects tagged with both tags, and the total number of objects in the group:

$$sim(t_i, t_j) = \exp[-\frac{\max(n_i, n_j) - n_{ij}}{\log |G| - \min(n_i, n_j)}] \tag{9}$$

where $n_i = \log |U(t_i)|$, $n_j = \log |U(t_j)|$ and $n_{ij} = \log |U(t_i) \cap U(t_j)|$. After each iteration q, the utility score of each tag is updated according to:

$$f_q(t_i) = z \cdot \sum_{t_j \in T(G)} f_{q-1}(t_j) \cdot p(t_j, t_i) + (1 - z) \cdot f_0(t_i) \tag{10}$$

with z being a weight parameter. The process is repeated until the utility score of each tag t converges to a value $f(t)$. Then, the top k tags with respect to $f(t)$ are selected to construct the tag cloud for the group.

4.2 Based on Diversity

A limitation of the approaches examined above is that the utility score of each tag is computed independently, without taking into consideration the rest of the tags in the cloud. This may result in cases where certain objects of the group are over-represented in the tag cloud, i.e., they have many of their tags appearing in the cloud, while other objects are under-represented, having very few or no tags in the cloud. Especially the latter case constitutes a very negative scenario, since it implies that these objects are not reachable via the tag cloud. This problem can be addressed by following a strategy that aims at increasing the *diversity* or *novelty* of the members of the tag cloud [6, 4]. In the following, we describe two approaches that can be used for this purpose.

4.2.1 Diversity

The goal in this case is to select tags that are as dissimilar as possible from each other, in the sense that appear in different sets of objects. Specifically, a tag has high utility, if there are no other tags already in the cloud that have a high similarity to it, i.e., that appear approximately in the same set of objects. This can be quantified by computing the minimum distance between the considered tag t and the currently selected tags in the cloud T_G. Since we do not wish to completely discard the criterion of frequency, we include it as an additional factor in computing the utility score of a tag. Specifically, the utility score of a tag t is defined as

$$f(t) = \lambda \cdot fr(t, G) + (1 - \lambda) \cdot (1 - \max_{t_i \in T_G} sim(t, t_i)) \tag{11}$$

where the parameter λ is used to weight the importance of frequency with respect to the factor of diversity. For example, assume that $\lambda = 0$. If a tag t appears in exactly the same set of objects as another tag t_i already selected in the cloud, then $sim(t, t_i) = 1$, and therefore $f(t) = 0$. On the contrary, if there is no tag in the cloud that appears in any of the objects tagged with t, then the maximum similarity of t with the current members of T_G is 0, and therefore the utility of t becomes equal to 1.

As can be seen by Equation 11, the selection of a tag now can not be done independently, but depends instead on the selection of other tags in the cloud. However, examining all possible combinations is clearly impractical due to the high computational cost. For this reason, a greedy approximation algorithm can be used, as is typically done in search results diversification. The algorithm is simple and proceeds as follows. First, the tag cloud is initialized with the tag that has the highest frequency. Then, the remaining $k - 1$ tags are selected by adding to the cloud, at each iteration, the tag that maximizes the utility value in Equation 11, with respect to the contents of the cloud up to that step.

4.2.2 Novelty

Another approach to diversify the members of a tag cloud is to emphasize on the novelty of newly selected tags, while the cloud is constructed. In document retrieval, this is formalized using the notion of *information nuggets*. An information nugget v is a piece of information contained in a document. Each document may contain one or more information nuggets, and the same information nugget may appear in more than one documents. When documents are retrieved, a document is selected if it contains as many previously unseen information nuggets as possible.

In our case, we consider each object to constitute one information nugget[1]. Selecting a tag t, provides a set of information nuggets $V(t)$ corresponding to the objects that this tag is assigned to. Under the above assumption for the definition of an information nugget, we have $V(t) = U(t)$. Let n_{v,T_G} be the number of times that a nugget v has already appeared in the tags currently contained in the cloud, i.e., $n_{v,T_G} = |\{t_i \in T_G : v \in V(t_i)\}|$. Also, let N_V be the total number of information nuggets in the group; in our case, $N_V = |G|$. Given a tag t and the current contents of the tag cloud T_G, the utility score of t is defined as

$$f(t) = \frac{\sum\limits_{v \in V(t)} \gamma(n_{v,T_G})}{N_V} \tag{12}$$

The function $\gamma()$ is a discount function that reduces the contribution of each information nugget of the tag t based on the number of times it has already been seen by previously selected tags. To maximize the emphasis on novelty, this function can be defined to return 1 if $n_{v,T_G} = 0$, and 0 otherwise. In that case, if, for example, a tag t appears only in a single object u, and there is already another tag of u in the cloud, then the utility score of t is 0.

As with Equation 11, in this definition of the utility function f the score of a tag depends also on the other tags existing in the cloud. Therefore, tags may again be selected using a similar greedy algorithm to the one described in the previous section. Specifically, the algorithm starts by selecting first the tag that provides the highest number of information nuggets. According to our definition of nuggets, where there is a one-to-one mapping between objects and information nuggets, this is the tag having the highest frequency. At each subsequent iteration, the tag providing the highest number of new nuggets is identified and added to the cloud, until a total number of k tags have been selected.

4.3 Based on Rank Aggregation

The tag ranking strategies presented so far take into consideration the criteria of frequency and novelty. However, there is another source of information that can be taken into account, when constructing the tag cloud of a group of objects: the order in which the tags appear in these objects. Although in typical applications there are no explicit semantics or criteria determining the order in which users assign tags to an object, it is reasonable to assume that there exists some (even small) correlation between this order and the relevance or importance of the tags for the target object. Alternatively, it is also possible to determine and use a different ranking for the tags of each object, according to a specified criterion, as in [14].

#	Method					
	Freq.	TF.IDF	R.W.	Div.	Nov.	Rank Aggr.
1	nikon	panoramio	nikon	nikon	nikon	nikon
2	people	wiki	d5000	people	people	wiki
3	d5000	ulisse	people	italy	love	wikipedia
4	italia	gps	italy	music	hotel	people
5	italy	wikipedia	italia	canada	ottawa	d5000
6	portrait	aci	flickr	portrait	macro	cc
7	flickr	basalt	creative	love	sky	gente
8	creative	faraglioni	portrait	d90	rome	panoramio
9	cc	cc	cc	2010	barcelona	portrait
10	common	fishermen	common	london	d300	sicilia

Table 1: Top 10 tags selected by the different algorithms for the Flickr group "This is nikon art".

To take the order of tags into account, we define a utility function based on the Borda Count method. This is a voting algorithm that has been shown to be optimal with respect to desired symmetry properties of typical election systems, and has also been used for metasearch in IR [1]. The Borda Count method works as follows. Each voter ranks a fixed set of c candidates in order of preference. For each voter, the top ranked candidate receives c points, the second ranked candidate receives $c - 1$ points, and so on. At the end, the candidates are ranked in order of their total points. The candidate with the most points wins the election.

In our case, voters correspond to objects and candidates correspond to tags. To take the ranking of tags into account, we discount the "vote" given by an object to a tag based on the position of this tag in the object's tag list, using a discount function as discussed for Equation 2. Thus, we define the utility score of each tag as follows:

$$f(t) = \frac{\sum\limits_{u \in U(t)} \alpha(r_u(t))}{|G|} \tag{13}$$

where $r_u(t)$ is the position (starting from zero) of the tag t in the list of tags assigned to object u. For example, if the tag t appears in the first position of the tag list of all the objects in the group, then $f(t) = 1$. The tag cloud comprises the top k tags ranked in decreasing order of their utility score.

5. EVALUATION METHODOLOGY

In the previous section, we have presented different approaches for selecting and ranking tags for a group of objects. Table 1 shows the top 10 tags selected by these methods for the Flickr group "This is nikon art". One can notice, for example, that frequency-based ranking selects tags like *flickr* or both *italy* and *italia*, which in other methods are replaced by other more intresting tags, such as *fishermen*, *music*, *hotel*, *rome*, *sicilia*, etc.

The question that arises is which of these tag clouds is better and why. In the following, we propose an evaluation methodology for comparing different tag clouds. The typical purpose of a tag cloud is to serve as a summarized representation of a group in order to allow users to search and navigate through its items, to drill down to specific items of

[1] A more fine grained definition for information nuggets is possible if the content of the objects is analyzed and taken into consideration; however, this is orthogonal to our work.

interest, and to receive recommendations for groups when adding a new item (e.g., [10, 14]). The methodology and metrics presented in this section aim at providing an objective and automatic way to compare the characteristics and effectiveness of different tag clouds with respect to these properties and tasks.

5.1 Metrics for Search and Navigation

In the following, we consider characteristics of a tag cloud that make it useful for search and navigation. Notice that similar metrics have been also proposed for evaluating faceted search interfaces (see Section 2 for more details).

5.1.1 Coverage

Since a tag cloud aims at providing an entry point for searching and navigating through the objects of a group, ideally every object should be reachable through the tag cloud. That is, for every object, at least one of its tags should appear in the tag cloud. Otherwise, any tag that the user may select would lead to a subset of results that does not contain this particular object. To measure this property, we define the *coverage* of a tag cloud T_G with respect to the group of objects G it represents as the portion of the objects in the group that have at least one tag appearing in the cloud, i.e.,

$$coverage(T_G) = \frac{|\{u \in G\ :\ T(u) \cap T_G \neq \emptyset\}|}{|G|} \tag{14}$$

The higher the coverage, the more effective the tag cloud is.

5.1.2 Overlap

Another aspect is that a tag cloud, similar to a (multi-) document summary, should avoid redundancy. This means that we would like to avoid cases where different tags in the cloud, when selected, lead to the same or very similar subsets of objects. To quantify this aspect, we measure the overlap between two tags t_i and t_j by computing the portion of objects tagged with t_j that are also tagged with t_i, i.e., $|U(t_i) \cap U(t_j)|\ /\ |U(t_j)|$. Assume, for example, that a user selects first t_i, browses the results, then goes back, and selects t_j. If the overlap between t_i and t_j is high, the second step will return little or no new results. Thus, we define the *overlap* of the tag cloud as the average overlap between each pair of tags:

$$overlap(T_G) = \frac{\displaystyle\sum_{t_i, t_j\, \in\, T_G} \frac{|U(t_i) \cap U(t_j)|}{|U(t_j)|}}{|T_G| \cdot (|T_G| - 1)} \tag{15}$$

The lower the overlap, the more effective the tag cloud is.

5.1.3 Selectivity

As already mentioned, a tag cloud should facilitate users to drill down to specific objects of interest. Selecting all the common tags between an object and the cloud provides the best "zoom in" that can be achieved for this object. The question then is how many other objects remain after this selection is made. Assume that a user is insterested in the object u, and that all the tags of u appearing in the tag cloud, i.e., $T(u) \cap T_G$, have been selected. The result list will then contain u, as well as all other objects that also have (at least) those tags. That is, the result list will comprise every object $u_i \in G$ such that $T(u) \cap T_G \subseteq T(u_i)$, while the rest of the objects will have been filtered out. The number of filtered out objects should be as high as possible. We call this number, normalized to the total number of objects, the selectivity of u with respect to the tag cloud T_G. Thus, we measure the *selectivity* of the tag cloud by computing the average selectivity of the objects in the group:

$$selectivity(T_G) = \frac{\displaystyle\sum_{u \in G} \frac{|\{u_i \in G\ :\ (T(u) \cap T_G) \nsubseteq T(u_i)\}|}{|G|}}{|G|} \tag{16}$$

The higher the selectivity, the more effective the tag cloud.

5.2 User Navigation Model

The metrics described so far allow to quantify different properties of a tag cloud with respect to search and navigation. However, they are "static", in the sense that they do not consider the (order and cost of) actions taken by a user when using the tag cloud to find items of interest. Next, we describe a user navigation model that can be used to evaluate different tag clouds in this respect.

The reasons for preferring a simulation model to an actual user study are the following. On the one hand, if the same user evaluates alternative tag clouds for a group, the results can not be easily compared, because, when using one tag cloud for navigating and searching for items, the acquired knowledge about which tags were more or less useful will affect the choices made when using the other tag clouds. On the other hand, if different users evaluate the alternative tag clouds for a group, again the results can not be easily compared, due to cognitive differences among the users or differences regarding what each user perceives as interesting or relevant. A possible solution would be to use non-overlapping, sufficiently large sets of users for evaluating alternative tag clouds, and then comparing the average results among those sets. However, this makes the user study even more expensive to conduct and to reproduce. Instead, the proposed simulation model provides an easy and objective way for comparison. Notice that such a methodology has also been followed in [9] to evaluate user interfaces for personalized and interactive faceted search instead of relying on actual user studies forsimilar reasons. To simulate user navigation, we assume that the user is trying to find items of interest in a group that are described by a set of tags T_{nav}. To find an item, the user can perform two types of actions: she can scan the list of results or she can select a tag in the cloud to reduce the size of the result set. Each of these actions is associated with a cost. The goal is to measure the total cost for finding an item.

Initially, the list of items presented to the user contains all the objects of the group, typically organized in pages of fixed size (e.g., 10 objects per page). Let c_p be the cost of scanning one page of objects. We also assume that the user is willing to scan up to a relatively small number of pages, say n_p; otherwise, if the list comprises more than n_p pages, she would prefer instead to select a tag to narrow down the list of results. To capture this, we set the cost c_t of selecting a tag to be equal to scanning n_p pages, i.e., $c_t = n_p \cdot c_p$.

The process works as follows:

1. The result list is initialized with all the objects in the group.

2. If the current result list contains at most n_p pages or

if there are no more tags left to use, the user scans the results and the session ends.

3. Otherwise, the user selects one tag from $T_{nav} \cap T_G$ and the result list is updated. The process repeats from step 2.

At the end of the session, the navigation cost is the sum of the costs of all the actions performed by the user (i.e., tag selections and page scans). This cost is divided by the cost needed to scan all the initial list of objects without using the tag cloud, to obtain a normalized value in $[0, 1]$. Therefore, if the user performed n_1 tag selections and n_2 page scans, then the navigation cost is:

$$nav\_cost = \frac{n_1 \cdot c_t + n_2 \cdot c_p}{n_0 \cdot c_p} \tag{17}$$

where $n_0 = \lceil (|G| \,/\, (page\_size) \rceil$.

There are two questions still to be answered to fully specify the simulation process. First, which are the tags T_{nav} that are used for the navigation. Second, when a user selects one tag from $T_{nav} \cap T_G$, which tag is selected.

Regarding the first question, for the purpose of the simulation, we assume that for each object u in the group there exists one user that wants to navigate the collection using the tags of this object, i.e., $T_{nav} = T(u)$, and we compute the average navigation cost among all users. Of course, in practice, more users may be interested in certain objects rather than others. In that case, it is possible to generalize the process using different weights for the users, as discussed also in Section 4.1.1.

For the second issue, we adopt the approach used in [9] for selecting facet-value pairs when simulating user behavior in a faceted browsing interface. Specifically, three types of users are considered. *First-match users* select the tag in $T_{nav} \cap T_G$ that has the highest rank in T_G. The intuition for this lies in the fact that tags in a cloud are visualized according to their ranking, with top tags being presented more prominently than others (typically, in terms of font size). Thus, users are more likely to spot and select highly ranked tags first. In contrast, *last-match users* select the tag having the lowest rank. The reason for this behavior is that those tags are less frequent, and therefore more selective, leading to a smaller subset of results. Finally, *random-match users* select a tag randomly.

5.3 Group Recommendation Accuracy

So far, we have considered the task of using the tag cloud to find items of interest within a group. Another important and common task is to recommend groups for new items. In this case, we assume that we are given an object u tagged with a set of tags $T(u)$, and the system needs to recommend groups for u from the set of groups $\mathcal{G}$ existing in the collection. A straightforward approach is to compare u with each object in each available group and recommend the group that contains the most similar objects to u. However, this is clearly very expensive, especially when this operation needs to be done online. Instead, we want to use the tag cloud of each group for this purpose.

To assign an object to a group, we compute a similarity score based on the common tags between the object and the tag cloud of the group, taking also into consideration the importance (i.e., the ranking) of these tags in the tag cloud. Specifically, we define this similarity as follows:

$$sim(u, G) = \frac{\displaystyle\sum_{t \in T(u) \cap T_G} \alpha(r(t))}{|T(u)|} \tag{18}$$

where $a(r(t))$ is a discount function applied on the rank $r(t)$ of the tag t in T_G, as in Equation 2.

Given an object u, groups are then recommended in decreasing order of similarity. Using different tag clouds, results in different lists of retrieved groups. In addition, if we assume that we know the relevant groups for the given object, i.e., the correct assignments, then it is possible to evaluate different tag clouds using standard IR metrics.

6. EXPERIMENTAL EVALUATION

We have conducted an empirical evaluation of the tag ranking algorithms presented in Section 4, using a large, real-world dataset obtained from Flickr containing groups of tagged images. First, we describe the characteristics of this dataset, and then we present the results of the evaluation with respect to the metrics presented in Section 5.

6.1 Dataset

To evaluate the tag ranking methods, we collected a dataset comprising groups of tagged photos from Flickr. We chose Flickr among a set of several popular Web 2.0 applications that provide tagging functionality and use tag clouds, such as Delicious, Technorati, Facebook or Last.fm, for mainly two reasons. First, there exists a very large number of groups of photos in Flickr, with groups often containing hundreds or thousands of photos, having several tags assigned to each of them. This allowed us to obtain a very large dataset in order to make the evaluation more interesting and challenging. Second, obtaining the data was significantly facilitated by the API[2] provided by Flickr for such purposes.

The Flickr API was used to collect data about groups of photos over a period of two weeks. Flickr does not provide a direct way to browse all available groups; instead, one needs to search for groups based on keywords. Thus, to obtain groups of photos we used a total of 10 keywords, in particular: *autumn*, *animals*, *architecture*, *city*, *beach*, *food*, *film*, *christmas*, *art*, and *car*. These were selected from the list of “all time most popular tags” in Flickr[3].

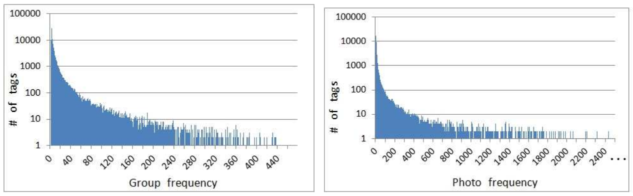
For each one of these keywords, we retrieved the top 60 groups returned by the API. Specifically, for each group, we retrieved up to 2000 photos, storing for each photo its ID, its owner, the date posted, the group(s) it belongs to, and the list of assigned tags (with the order they appear in Flickr). Note that in some cases, some groups or photos could not be retrieved, either because of being set as private or due to some connection error. Thus, there exist variations in the number of groups that were finally collected per keyword, as well as the number of photos per group, which however corresponds to a realistic scenario. Moreover, some pre-processing was done on the collected data to reduce noise, including stopword removal and stemming. After this process, we also removed: tags that had been used by less than 2 users; photos that had less than 4 tags; and groups that contained less than 50 photos.

[2] `http://www.flickr.com/services/api/`

[3] `http://www.flickr.com/photos/tags/`

Keyword	# of groups	# of photos per group			# of distinct tags per group			# of tags per photo		
		avg	max	min	avg	max	min	avg	max	min
animals	56	1248	1758	843	3068	5106	1114	10	142	4
architecture	59	1416	1748	510	2865	5245	579	12	71	4
art	55	1164	1650	418	3098	5820	1232	12	73	4
autumn	29	1220	1620	519	2655	5361	134	11	69	4
beach	38	1216	1548	426	2175	4581	616	13	72	4
car	48	1391	1719	973	2625	5262	1027	13	92	4
christmas	15	844	1438	417	1644	3438	605	10	68	4
city	49	1278	1865	415	2586	4331	581	11	72	4
film	53	1367	1862	460	2758	4407	1034	12	72	4
food	49	1212	1741	559	2578	4387	1070	10	70	4
TOTAL	451	1270	1865	415	2707	5820	134	11	142	4

Table 2: Dataset statistics.



(a) (b)

Figure 2: Distribution of (a) group frequency and (b) photo frequency of tags.

The final dataset contained 451 groups, 488,112 photos, and 112,514 tags, with each group having on average 1270 photos and 2707 distinct tags. Table 2 provides detailed statistics about the dataset. Moreover, Figure 2 plots the distribution of tag frequencies, in terms of number of groups and number of photos they appear in. For example, one can see that the majority of the tags has a relatively low group frequency, e.g., less than 30 groups, while there are only a few tags that appear in more than half of the groups. The distribution is similar for the photo frequency, with most tags appearing in not more than 100 different photos. Of course, there are also some tags like names of colors, cities or countries, or tags such as *sky*, *sea*, *street*, that appear in the majority of the groups and in more than 10,000 photos.

6.2 Results

We now present the results of our experiments on the Flickr dataset described above. We have implemented the algorithms described in Section 4, and we compare them applying the evaluation methodology presented in Section 5.

In the implementation and experiments, we used the following configuration. For the random walk algorithm (Equation 10) and for the diversification algorithm (Equation 11), we used $z = 0.5$ and $\lambda = 0.5$, respectively. As discount function in Equation 2 we used the one in Equation 3 with $\beta = 0.1$. In Equation 12, we set $\gamma(n) = 1$ for $n = 0$, or 0 otherwise, to emphasize novelty. Finally, in the user navigation model, we used $n_p = 2$ and we assumed that each page of results displays (at most) 10 objects.

In the following discussion and in the plots, we refer to the methods presented in Section 4 using the abbreviations FRQ, TFIDF, RW, DIV, NOV, and RA, respectively. The presented results refer to average values over all the 451 groups in the dataset.

6.2.1 *Coverage, Overlap and Selectivity*

Figure 3 displays the results for the different tag ranking algorithms with respect to the metrics for search and navigation presented in Section 5, and for different values of the number of tags k in the cloud.

As shown, FRQ performs reasonably well, mainly for the metrics of coverage and selectivity and less for overlap. For example, it provides a coverage starting from 87% for top-k = 20, reaching 95%, for top-k = 100. RW and RA exhibit a similar performance to FRQ, with the latter outperforming the other two regarding overlap. In contrast, the performance of TFIDF is very low both for coverage and selectivity, but it outperforms all three previous approaches in overlap. Finally, DIV and NOV perform better than all other approaches. Especially for coverage, NOV achieves excellent results, starting from 93% for top-k = 20 and reaching 99% for top-k = 100. It also has good results for overlap for all values of k, although for $k > 60$ DIV becomes equally good or even better.

As expected, increasing the size of the tag cloud improves the performance of all methods in all metrics.

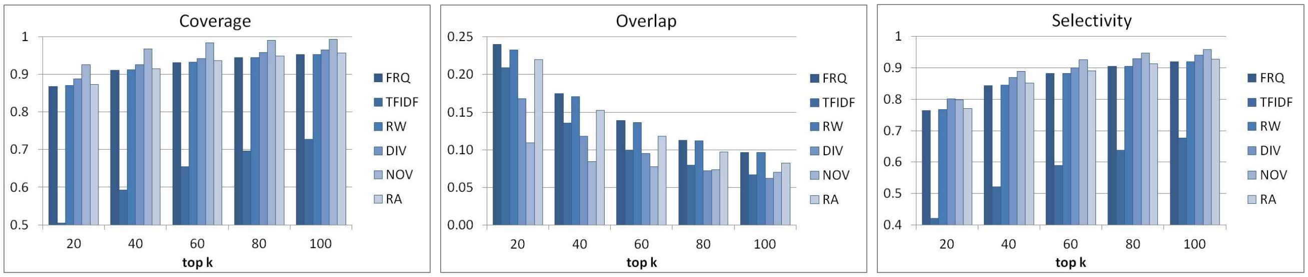
6.2.2 *Navigation Cost*

Figure 4 displays the navigation cost for the tag clouds created with the different ranking algorithms, considering the three types of users described in Section 5.2, and for different tag cloud sizes.

A first observation is that TFIDF has a much higher navigation cost than all other methods. This is a direct consequence of its low coverage, since photos that are not covered by the tag cloud can not benefit from it during navigation. For the rest of the methods, the observations are similar to the results presented above. This has to do with the fact that the navigation cost is affected by coverage and selectivity. Thus, for example, since NOV performed better than all other methods in both those metrics, this also contributed to having the lowest navigation cost. As expected, the navigation cost decreases for all methods as the tag cloud size increases; also, it is higher for first-match users and lower for last-match users.

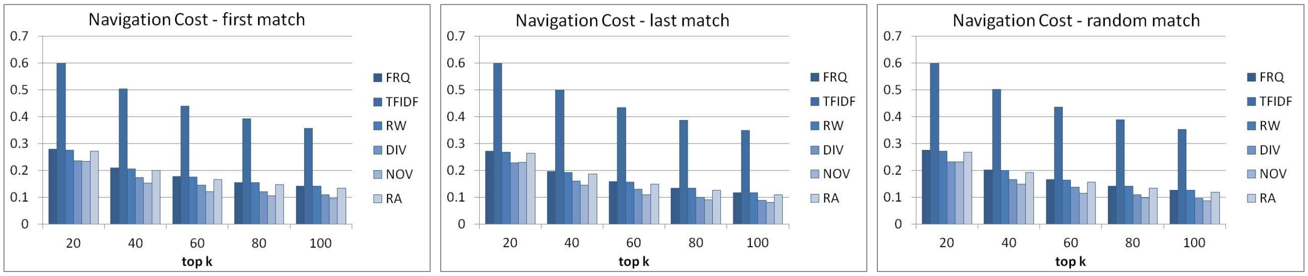
6.2.3 *Recommendation Accuracy*

Here we consider the task of group recommendation. Given a tagged photo and a set of groups, the goal is to recommend groups for this photo based on their tag clouds. For each of the 451 group, we selected as test photos the 10 most recent ones added to it by different users. Then, for each test



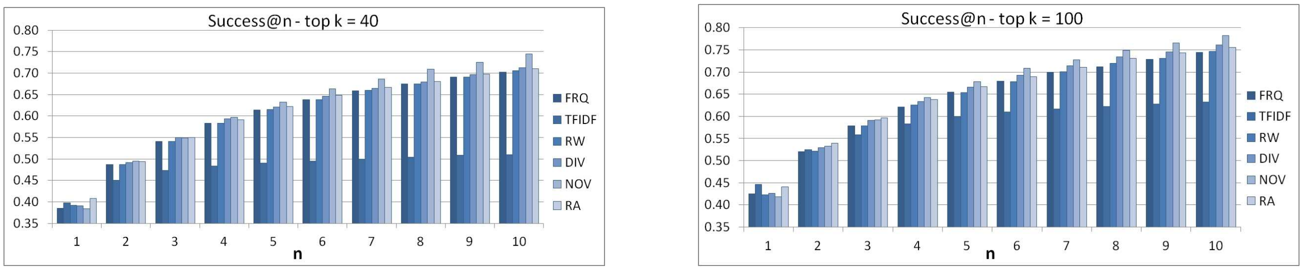
(a) coverage (b) overlap (c) selectivity

Figure 3: Evaluation results for the metrics (a) coverage (b) overlap and (c) selectivity.



(a) first-match (b) last-match (c) random-match

Figure 4: Navigation cost for (a) first-match (b) last-match and (c) random-match users.



(a) top-k = 40 (b) top-k = 100

Figure 5: Success@n for group recommendation with (a) top-k = 40 and (b) top-k = 100.

photo, we computed a ranked list of recommended groups, using the different tag clouds provided by the ranking algorithms. The similarity between a photo and the tag cloud of a group was computed according to Equation 18.

The performance of each tag cloud is determined by how high the relevant group(s) of a photo appear in the list of recommendations. As relevant, we considered the group from which the photo was selected, as well as any other groups it might also belong to. Since very few photos appear in more than one group, we used the common IR metric *success*@n, which takes the values 1 or 0, depending on whether a relevant result is found among the top n recommendations.

Figure 5 displays the average values of *success*@n over all the test photos, for different values of n and for tag clouds of size 40 and 100 tags. Interestingly, when only the first recommendation is considered, TFIDF and RA achieve the highest accuracy. However, the performance of TFIDF remains approximately at the same levels as n increases. This is attributed to the fact that TFIDF discards many frequent tags. This helps to make a more accurate recommendation for photos that are covered by the tag cloud, but it misses for all the rest. Instead, RA and DIV continue to be better for n = 2 or 3. For higher values of n, NOV achieves the best performance.

6.2.4 *Execution Time*

Finally, we also measured the execution time for the different ranking algorithms. The following table shows the average time (in sec) per group, for top-k = 100:

FRQ	TFIDF	RW	DIV	NOV	RA
0.002	0.003	87.813	9.552	0.296	0.008

FRQ, TFIDF and RA are simple, one-pass algorithms, therefore their execution time was very low. The diversification algorithms are more expensive, since they need to perform k passes. At each pass, each of the remaining tags is compared either with each tag already in the cloud (DIV) or with the set of information nuggets in the cloud (NOV). Thus, DIV requires even more computations than NOV. Indeed, the execution time of NOV was below half second, while DIV required a few seconds. Finally, RW exceeded 1 min, which makes it not suitable for online computation.

6.2.5 *Summary of Results*

Summarizing the results, we can make the following main observations:

- As a baseline, FRQ achieved reasonably good results, which verifies that it is a good criterion for selecting tags. Nevertheless, it was outperformed by other methods in all metrics.
- TFIDF showed a very poor performance for all metrics except overlap. This is because it penalizes frequent tags that occur also in many groups. Although this lowers the overlap also within each group, excluding those tags results in many photos not having any common tags with the cloud.
- RW failed to show a noticeable improvement over FRQ. This means that, in our setting, propagating the scores in the tag graph did not improve the ranking. Even worse, it significantly increased the computational cost. Still, it would be worth investigating whether enriching the tag graph with content-based similarity of tags and photos provides a more positive effect.
- The diversification methods, DIV and NOV, achieved the best performance in all metrics. NOV clearly outperformed all methods, especially for coverage, as well as for recommendations. This is due to the fact that it selects tags that cover as many not previously seen items as possible. In some cases, the performance of DIV was equally or slightly better than NOV; however, NOV has also a lower computational cost.
- Finally, RA performed overall a little better than FRQ but not as well as DIV and NOV. Also, its execution time was similar to frequency.

7. CONCLUSIONS

In this paper, we have presented a set of methods and metrics for constructing and evaluating tag clouds to describe groups of tagged resources. The presented algorithms include frequency or *tf.idf* based ranking, random walk on tag graphs, diversification of tags, and rank aggregation. The proposed metrics cover several aspects of tag clouds for search, navigation and recommendations. The results of our large-scale evaluation on groups of Flickr photos have shown that methods employing diversification or rank aggregation can improve the performance of tag clouds with respect to these metrics, compared to the traditional frequency-based ranking, while still having a similar or comparable computation time and without relying on content-based analysis. There exist several interesting directions for future work, as already pointed out while discussing related efforts. These include extracting semantics of tags and exploiting content-based similarity of objects.

8. REFERENCES

[1] J. A. Aslam and M. H. Montague. Models for metasearch. In *SIGIR*, pages 275–284, 2001.

[2] A. Budura, S. Michel, P. Cudré-Mauroux, and K. Aberer. To tag or not to tag - harvesting adjacent metadata in large-scale tagging systems. In *SIGIR*, pages 733–734, 2008.

[3] R. Cilibrasi and P. M. B. Vitányi. The Google similarity distance. *IEEE Trans. Knowl. Data Eng.*, 19(3):370–383, 2007.

[4] M. Drosou and E. Pitoura. Search result diversification. *SIGMOD Record*, 39(1):41–47, 2010.

[5] K. Fujimura, S. Fujimura, T. Matsubayashi, T. Yamada, and H. Okuda. Topigraphy: visualization for large-scale tag clouds. In *WWW*, pages 1087–1088, 2008.

[6] S. Gollapudi and A. Sharma. An axiomatic approach for result diversification. In *WWW*, pages 381–390, 2009.

[7] Z. Guan, J. Bu, Q. Mei, C. Chen, and C. Wang. Personalized tag recommendation using graph-based ranking on multi-type interrelated objects. In *SIGIR*, pages 540–547, 2009.

[8] M. A. Hearst. Clustering versus faceted categories for information exploration. *Comm. ACM*, 49(4):59–61, 2006.

[9] J. Koren, Y. Zhang, and X. Liu. Personalized interactive faceted search. In *WWW*, pages 477–486, 2008.

[10] B. Y.-L. Kuo, T. Hentrich, B. M. Good, and M. D. Wilkinson. Tag clouds for summarizing web search results. In *WWW*, pages 1203–1204, 2007.

[11] C. Li, N. Yan, S. B. Roy, L. Lisham, and G. Das. Facetedpedia: dynamic generation of query-dependent faceted interfaces for wikipedia. In *WWW*, pages 651–660, 2010.

[12] X. Li, C. G. M. Snoek, and M. Worring. Learning social tag relevance by neighbor voting. *IEEE Trans. on Multimedia*, 11:1310–1322, Nov. 2009.

[13] D. Liu, X.-S. Hua, M. Wang, and H.-J. Zhang. Retagging social images based on visual and semantic consistency. In *WWW*, pages 1149–1150, 2010.

[14] D. Liu, X.-S. Hua, L. Yang, M. Wang, and H.-J. Zhang. Tag ranking. In *WWW*, pages 351–360, 2009.

[15] S. Lohmann, J. Ziegler, and L. Tetzlaff. Comparison of tag cloud layouts: Task-related performance and visual exploration. In *INTERACT (1)*, pages 392–404, 2009.

[16] L. Lovász. Random walks on graphs: A survey. *Combinatorics Paul Erdos is Eighty*, 2(1):1–46, 1993.

[17] E. Oren, R. Delbru, and S. Decker. Extending faceted navigation for RDF data. In *ISWC*, pages 559–572, 2006.

[18] T. Rattenbury, N. Good, and M. Naaman. Towards automatic extraction of event and place semantics from Flickr tags. In *SIGIR*, pages 103–110, 2007.

[19] S. B. Roy, H. Wang, G. Das, U. Nambiar, and M. K. Mohania. Minimum-effort driven dynamic faceted search in structured databases. In *CIKM*, pages 13–22, 2008.

[20] J. Schrammel, M. Leitner, and M. Tscheligi. Semantically structured tag clouds: an empirical evaluation of clustered presentation approaches. In *CHI*, pages 2037–2040, 2009.

[21] G. Smith. *Tagging: People-powered Metadata for the Social Web*. New Riders Publishing, Thousand Oaks, CA, USA, 2008.

[22] P. Venetis, G. Koutrika, and H. Garcia-Molina. On the selection of tags for tag clouds. In *WSDM*, pages 835–844, 2011.

[23] L. Wu, L. Yang, N. Yu, and X.-S. Hua. Learning to tag. In *WWW*, pages 361–370, 2009.