Social Recommender Systems for Web 2.0 Folksonomies

Stefan Siersdorfer
University of Hannover
L3S Research Center
siersdorfer@l3s.de

Sergej Sizov
University of Koblenz
ISWeb Research Group
sizov@uni-koblenz.de

ABSTRACT

The rapidly increasing popularity of Web 2.0 knowledge and content sharing systems and growing amount of shared data make discovering relevant content and finding contacts a difficult enterprize. Typically, folksonomies provide a rich set of structures and social relationships that can be mined for a variety of recommendation purposes. In this paper we propose a formal model to characterize users, items, and annotations in Web 2.0 environments. Our objective is to construct social recommender systems that predict the utility of items, users, or groups based on the multi-dimensional social environment of a given user. Based on this model we introduce recommendation mechanisms for content sharing frameworks. Our comprehensive evaluation shows the viability of our approach and emphasizes the key role of social meta knowledge for constructing effective recommendations in Web 2.0 applications.

Categories and Subject Descriptors

H.4.0 [**Information Systems**]: Information Systems Applications—*General*

General Terms

Algorithms, Experimentation, Human Factors

1. INTRODUCTION

Popularity and data volume of modern Web 2.0 content sharing applications originate in their ease of operating for even unexperienced users, suitable mechanisms for supporting collaboration, and attractiveness of shared annotated material (images in Flickr, videos in YouTube, bookmarks in del.icio.us, etc.). Despite disagreement on the exact definition of Web 2.0, it is common to find community and collaboration as key concepts in this latest online phenomenon. Increasingly, online content is being created, edited and shared by whole communities of users, demonstrated by the popularity of applications such as Flickr, YouTube, and Del.icio.us [1]. Web 2.0 applications provide a rich set of structures and annotations that can be mined for a variety of purposes. For example, Flickr postings are accompanied with a variety of descriptive metadata, such as creator (and/or owner), a textual description, thematic tags, temporal and geographic information, and comments by other Flickr users on specific regions of uploaded pictures. Using these structures, a variety of relationships between users, tags, pictures, and groups can be explored.

1.1 Motivation

The growing size of folksonomies poses new challenges in terms of search and mining for relevant content and finding other users sharing the same interests. Ideally, a Web 2.0 platform should provide the user with adaptive browsing mechanisms and recommendations for potentially relevant content, users, or annotations. This functionality clearly goes beyond the location of matching items for a keyword-based query and poses a new level of Web 2.0 exploration service. A challenging research issue is therefore the development of suitable recommendation methods.

In many aspects, recommendation algorithms for folksonomies may substantially differ from methods known from the Web IR scenario. Web retrieval primarily utilizes the content of hypertext documents and the link structure of cross-references between them. In contrast, Web 2.0 systems provide much richer, collaboratively edited social metadata (comments, users and user groups, cross-annotations, etc.), which makes them more suitable for sharing of multimedia content. However, particular dimensions (e.g. annotations of the given photo or video) tend to be extremely sparse. In particular, this holds for explicit ratings of shared resources by different users. This issue (known as the ramp-up problem [15]) requires, in contrast to existing recommender systems, the use of implicit ratings which can be obtained through social relationships between users and resources, such as favorite lists.

The recommender system should take into account a specialized model of dependencies between users, items, and annotations that provides a good fit for observed properties of the folksonomy. Beyond these basic structures, modern Web 2.0 folksonomys contain additional features reflecting the social nature of the content sharing framework such as contacts, personal favorites, comments, groups, etc. In this article we consider Flickr as a prominent showcase for these social links.

[1] http://www.flickr.com, http://www.youtube.com, http://del.icio.us

HT'09, June 29–July 1, 2009, Torino, Italy.

1.2 Contribution

The main contributions of this article are the following:

- We develop and formalize a systematic and comprehensive framework for applying known Information Retrieval and recommender techniques to Web 2.0 structures. Our emphasis lies on representing and utilizing multiple social dimensions of the Web 2.0 environment, including common relationships between users, tags, and resources, and further aspects like favorite lists, contacts, user groups, or comments. We demonstrate how existing recommender techniques can be adapted and applied to this new application scenario.
- We describe a novel evaluation technique based on the reconstruction of existing social structures in Web 2.0 systems (e.g. favorites or contacts) which allows for large scale experimental evaluations without comprehensive human interaction.
- We provide a large-scale experimental study for photo and contact recommendations on Flickr comparing a variety of object representations, and showing the viability of our approach.

The rest of this article is organized as follows. We describe related work on Web 2.0 folksonmies, social networks and recommender systems in Section 2. In Section 3 we formalize our notion of the vectors space model for Web 2.0 environments, and, based on this formalization and additional social links, introduce our recommender framework in Section 4. Section 5 explains our evaluation methodology and shows results of systematic experiments on realistic large-scale data gathered from the Flickr folksonomy. Section 6 concludes and shows directions of our future work.

2. RELATED WORK

Schmitz et al. have formalized folksonomies and discuss the use of association rule mining for analyzing and structuring them in [25]. The recent work on folksonomy-based web collaboration systems includes [7], [10], and [18] which provide good overviews of social bookmarking tools with special emphasis on folksonomies. In [20], a model of semantic-social networks for extracting lightweight ontologies from del.icio.us is defined. In contrast to (rather application-specific) existing formalisms from these contributions, we introduce the generalized IR-like notion of a vector space that allows for representation of various social relationships between users and resources in folksonomies.

The analysis of topological properties is well-known in the areas of complex networks [23, 21, 2] and social network analysis (SNA). Typical examples of such measures are the clustering coefficient and the characteristic path length in the tripartite undirected hypergraph capturing relationships between users, annotations, and items. An equivalent common view on folksonomy data is known in Formal Concept Analysis [30, 9] as a *triadic context* [17, 26]. In many cases, suitable recommendations can be obtained by analyzing link-based authority measures of the folksonomy. A node ranking procedure for folksonomies, the FolkRank algorithm, has been introduced in [13]. FolkRank operates on a tripartite graph of users, tags and resources, and generates a ranking of tags for a given user. Another procedure is the Markov Clustering algorithm (MCL) in which a renormalization-like scheme is used in order to detect communities of nodes in weighted networks [28]. In contrast to exploitation of topological network properties, our approach aims to adopt vector based representations and methods to the specific aspect of social relationships in folksonomies.

The problem of evaluation methodology for recommender systems is systematically analyzed in [11]. In contrast to general evaluation methods presented there, our novel evaluation methodology aims to exploit implicit sources of relevance in the folksonomy using social relationships. Following this idea, we construct large-scale experiments that can be evaluated without human interaction.

Kleinberg [14] summarizes several different approaches to analyzing online information streams over time and detecting trends. His text mining scenario requires focusing on words that are neither too frequent nor too infrequent. The aspect of folksonomy dynamics is not directly addressed by methods presented in our article. In the future, we will extend our multi-dimensional representation model in order to capture trends and significant changes using dynamic tensor analysis methods [27].

Common recommender systems are usually used in one of two contexts: (1) to help users locate items of interest they have not previously encountered, and (2) to judge the degree of interest a user will have in item they have not yet rated. With the growing popularity of on-line shopping, E-commerce recommender systems have matured into a fundamental technology to support the dissemination of goods and services [24]. Much research has been undertaken to classify different recommendation strategies [4, 11], but here we divide them broadly into two categories: *Collaborative* and *Content-based* recommendations.

Content-based recommendation represents the culmination of efforts by the information retrieval and knowledge representation communities. A set of attributes for the items in a system is determined, such as terms and their frequencies for documents in a repository, so the system can build a profile for each user based on the attributes present in the items that a user has rated highly. The interest a user will have in an unrated item can then be deduced by calculating its similarity to their profile based on the attributes assigned to the item. In a collaborative recommender system, the ratings a user assigns to items is used to measure their commonality with other users who have rated the same or similar items. The degree of interest for an unseen item can be deduced for a particular user by examining the ratings of their neighbors.

Such systems are not without their deficiencies, the most prominent of which arise when the space of user ratings for items is sparsely populated or new items and users are added to the system - commonly referred to as the *ramp-up* problem [15]. *Hybrid* recommender systems, using a mixture of collaborative and content based approaches, have been developed to overcome some of these problems and to provide more robust systems. More recent recommender systems have also investigated the use of ontologies to represent user profiles [19]. Our approach can be seen as an integration of basic ideas from the hybrid methodology, with an emphasis on social relationships and dependencies.

To the best of our knowledge, this article is the first to describe the application and evaluation of recommender systems on social Web 2.0 structures.

3. A VECTOR SPACE MODEL FOR WEB 2.0 FOLKSONOMIES

In this Section, we provide a formal description of a vector space model comprising basic objects that commonly occur in most Web 2.0 environments. The vector representations obtained, together with additional social structures, will form a basis for our recommender techniques described in the next Section 4.

The structure of a content sharing framework is usually seen as a tripartite network [16] with ternary relations (tag assignments) between users $u \in U$, resources (e.g. images, media files) $r \in R$ and associated tags (arbitrary text labels, in our case) $t \in T$. The set of all relations of the content sharing framework is therefore $Y \subseteq U \times T \times R$ [25]. In this section, we transform this graph notation into a vector space model for characterizing basic folksonomy elements and the relationships between them.

3.1 Folksonomy clouds

Elements from U, R, or T in a content sharing framework can be mutually characterized through existing relationships between them. For instance, tags can be characterized by the resources they annotate and by the users that assign them. Analogously, users can be characterized by their resources and frequently used tags. We use these relationships (e.g. tags assigned by the user to a particular resource) and global statistics (e.g. fraction of user items annotated by a certain tag) for constructing characteristic feature vectors. For this purpose, we consider arbitrary subsets of Y coined *folksonomy clouds*. A folksonomy cloud is defined as $Y^* \subseteq Y$ and represents a context-dependent (or problem-dependent) subset of the relevant relations.

3.2 Tag-based Feature Vectors

The combination of term frequency and inverse document frequency $tf \cdot idf$ is commonly used in information retrieval for weighting terms of text documents. Following a similar motivation, we introduce the notion of *item-to-item frequency* (if) and an *inverse item frequency* (iif) for the feature vector of a context-dependent folksonomy cloud Y^*.

DEFINITION 3.1. *Let $u' \in U$ be an arbitrary user and Y^* a folksonomy cloud. The item-to-item frequency of u is defined as*

$$if(u) = |\{(u,t,r)\}|, (u,t,r) \in Y^* \wedge u = u' \tag{1}$$

Basically, if generalizes the notion of the well-known term frequency (tf), known from text retrieval, for a higher dimensional problem setting.

DEFINITION 3.2. *The inverse item frequency $iif(u)$ is defined as the ratio between cardinalities of sets T, R and their subsets T^*, R^* that have a relation with u in Y^**

$$iif(u) = \left(\log \frac{|T|}{|T^*|}, \log \frac{|R|}{|R^*|}\right) \tag{2}$$
$$with \quad T^* \subseteq T, R^* \subseteq R :$$
$$t^* \in T^* \Leftrightarrow \exists r : (u, t^*, r) \in Y^*$$
$$r^* \in R^* \Leftrightarrow \exists t : (u, t, r^*) \in Y^*$$

Analogously, iif adopts the idea of inverse document frequency (idf) from text retrieval and generalizes it for a multi-dimensional problem setting.

DEFINITION 3.3. *The overall weight $weight(u)$ for the user u in the cloud-specific feature vector of Y^* is defined as the L1-norm of the corresponding $if \cdot iif$ vector:*

$$weight_{Y^*}(u) = \|if(u) \cdot iif(u)\|_1 \tag{3}$$

The features for elements $t \in T^*$ and $r \in R^*$ are constructed analogously.

To allow for a more flexible construction of feature vectors, one may extend the definition (3.3) by arbitrary weighting coefficients α_m for particular dimensions of the feature vector (e.g. using $\alpha_m \in [0..1]$, $\sum \alpha_m = 1$) and re-defining:

$$iif_m^{\alpha}(i) \quad = \quad \alpha_m \cdot iif_m(i) \tag{4}$$

or additional smoothing/dampening normalizations. In the context of this article we restrict ourselves to feature vectors constructed according to definition 3.3.

For computing the similarity between feature vectors v_1 and v_2 we use the common notion of IR-style cosine measure:

$$sim(v_1, v_2) = \frac{v_1 \cdot v_2^T}{||v_1|| \cdot ||v_2||} \tag{5}$$

3.3 Latent Concept based Feature Vectors

The introduced tag-based tf*idf approach adopts the representational model from text IR. However, unlike text documents, tags of shared resources are extremely sparse. For this reason, in typical Web 2.0 environments the recommendation process suffers from the so called *vocabulary mismatch* problem [8], i.e. the problem that relevant resources might be annotated by semantically related but different tags and are ranked inadequately low. In the worst case of complete mismatch, such candidates are regarded as orthogonal to the user's interests. To overcome this shortcoming, the recommendation process can be combined with known dimensionality reduction techniques, e.g. latent semantic indexing (LSI) [6], probabilistic LSI (pLSI) [12], or latent Dirichlet allocation (LDA) [3].

As an instance of the mentioned models, we consider the LDA based generative probabilistic model in the context of this paper. The basic idea of this approach is to abstract from particular tags and to represent resource annotations and/or user profiles by mixtures over latent topics $z_1..z_k$ (i.e. hidden cloud-specific themes of interest), whereby each latent topic is characterized by a fixed conditional distribution over folksonomy tags. LDA assumes that all tags of resources (both observed and previously unseen) are generated by randomly chosen latent topics. In contrast to the singular value decomposition approach used in LSI, LDA has a well founded probabilistic background and tends to result in more flexible model fitting [3]. In contrast to the unigram mixture model [22], it allows resources to belong to multiple latent topics with different degrees of confidence. Unlike pLSI, LDA offers a natural way of assigning probabilities to previously unseen resources (i.e. resources with new, previously unseen combinations of tags). Beyond this, the number of estimated parameters in pLSI grows linearly with the number of training documents, which is prohibitively high for a large-scale folksonomy (e.g. Flickr) with millions of shared resources and thousands of users.

In line with [3], the annotation for the particular resource is generated by selecting a multinomial distribution over topics given the Dirichlet prior. For each tag, a topic is generated from the resource-specific topic distribution, and then a

Topic 1		Topic 2	
railroad	0.223	sky	0.098
trains	0.156	clouds	0.094
train	0.069	water	0.055
csx	0.050	trees	0.045
emd	0.028	sunset	0.044
ge	0.020	sun	0.035
tracks	0.012	river	0.021
amtrak	0.011	night	0.020
unionpacific	0.009	morning	0.020
sd402	0.007	snow	0.019
Topic 3		**Topic 4**	
child	0.230	myanmar	0.149
children	0.131	burma	0.120
girl	0.069	fergus	0.035
kid	0.036	woo	0.033
youth	0.024	macdonald	0.031
boy	0.020	bagan	0.018
fish	0.017	travels	0.014
mother	0.011	inlelake	0.013
love	0.011	yangon	0.011
johan	0.010	mandalay	0.010

Table 1: Characteristic features for sample topics of the Flickr dataset

tag is generated from the discrete distribution for that topic as follows:

1. The number of tags assigned to resource is chosen: $n \sim Poisson(\xi)$
2. The tag generating parameter is chosen: $\theta \sim Dir(\alpha)$
3. For each of the tags $t_i, i = 1..n$:
   - The generative topic for t_i is chosen: $z_i \sim Multinomial(\theta)$.
   - The tag t_i is generated using a multinomial probability with parameter β conditioned on z_i: $p(t_i|z_i, \beta)$

For the recommendation scenario, our application of LDA to an arbitrary community-centric folksonomy cloud $Y^* \subseteq Y$ can be summarized as follows. In the first step, for each community member $u \in U^*$ we construct the corresponding user-specific multiset of used tags $t \in T^*$. These sets are considered as 'training documents' and used for fitting the community-level properties α and β which are estimated using the variational EM procedure [3]. In the process of community analysis we also obtain the user-level variables θ, sampled once per user. As a result, we obtain the posterior distribution of the hidden topics $z_1..z_k$ given a user u:

$$p_u(\theta, \vec{z}|\vec{t}, \alpha, \beta) = \frac{p_u(\theta, \vec{z}, \vec{t}|\alpha, \beta)}{p_u(\vec{t}|\alpha, \beta)} \quad (6)$$

The user-specific distribution of hidden topics can be seen as a feature vector of dimensionality k. Analogously, the a posteriori topic distribution can also be estimated for tag combinations of particular resources. The similarity between user-specific feature vectors (e.g. using the common cosine similarity measure) can be considered as a similarity measure between users. Analogously, by comparing the user feature vector with feature vectors of particular resources we obtain the ranked list of content recommendations.

Alternatively, an individual LDA model can be constructed for a particular user. In this case, annotations of particular resources can be treated for training as separate documents. By choosing user-characteristic resources along social dimensions of the folksonomy (e.g. user favorites, commented resources, postings of friends which are on the user's personal contact list, etc.) we obtain a variety of alternate topical models which can be exploited for recommender scenarios.

Table 1 shows the top-10 tags for some of the multinomial distributions $p(t|z)$ for our Flickr data set (Section 5).

4. RECOMMENDER SYSTEMS FOR WEB 2.0

In this section, we show how existing concepts from recommender systems can be applied to Web 2.0 applications. We concentrate on Flickr as a prominent example; however, the proposed techniques carry forward to other Web 2.0 scenarios. Our recommender methodology builds on the vector representations described in the previous Section 3 in combination with additional social links obtained from folksonomy features such as contacts, comments, favorites, etc.

Given a large data set, the objective of a recommender system is to propose a subset of relevant or 'interesting' items from this data set to a user. In folksonomies such as Flickr these items can be photos, groups, or other users. This leads to recommendations such as:

- Given a user, recommend photos which may be of interest.
- Given a user, recommend users they may like to contact.
- Given a user, recommend groups they may want to join.

In the remainder of this section we will first provide a formal notion of recommender systems and show how it can be applied to a scenario such as Flickr. We will then discuss two approaches to tackle the recommender problem: content-based methods, and collaborative methods using social relationships. We will use notions based on a recent survey on recommender systems [1].

4.1 Problem Formalization

In order to formalize the relevance of an item with respect to user interests, we consider a utility function

$$ut : U \times S \rightarrow L \quad (7)$$

where U is a set of users, S a set of items, and L a set of relevance values (e.g. real values in $[0, 1]$).

The objective of a recommender system is to choose for a user $u \in U$ an item $s'_u \in S$ that maximizes the user's utility:

$$s'_u = argmax_{s \in S} ut(u, s) \quad (8)$$

More generally, we consider a ranked list of items with the highest utility values. Usually, the utility values are just known for a limited subspace of $U \times S$ (i.e., for those items rated by the user); thus, ut must be *estimated* for other elements of $U \times S$.

In the simplest case, for Flickr U corresponds to the set of Flickr users. There are extensions and generalizations possible: U can alternatively consist of tuples $(user, photo)$,

meaning a user viewing (or commenting on) a photo is provided with a list of other related photos. Since the result of this recommendation depends upon the photos, as well as the user's profile, this is an example of "personalization".

The set of items S can correspond to the set of photos (likely the most obvious option), the set of other Flickr users, the set of groups, or tags/concepts in Flickr.

4.2 Utility Assignments

An important issue is the estimation of appropriate utility values $ut(s,u)$ for a subset $U \times S$ of users and items; these utility values can be considered as "training data" for recommender methods. In classical recommender systems, direct relevance assignments from users are available, for instance, in form of a "star"-rating. In the movie application MovieLens.org, for example, users assign ratings to films according to a scale from 0 to 5. In Flickr, and many other Web 2.0 applications, such direct ratings are not available[2]. However, annotations supplied by users can be considered as *implicit ratings*. We exploit the following properties for resources (photos in our case):

- The photo belongs to the user. In this simple case we might assume that the user is interested in the photos that he has uploaded. To obtain a more fine-grained measure, the length of the textual description of the photo and the number of tags could be taken into account (the intuition behind this is that users will put more effort into the annotation of photos that are interesting to them).

- The user has marked the photo as a favorite. This is probably the most direct positive relevance assignment possible in Flickr, and is an explicit expression of interest in the photo.

- The user writes a comment about the photo. This implies that for the user, it was worth the effort of making a statement about the photo (whether positive or negative). More enhanced methods could take the length and date of the comment into account, and use sentiment classification to categorize the comment as positive or negative.

In our experiments, we use binary utility functions for each of these photo properties (i.e. $ut(u,s) = 1$ if the property holds for the given photo, and 0 otherwise). These utilities can be combined using a weighted linear combination of the utility values obtained for the different properties.

For assigning utility values to users (i.e. users are items and subject of recommendations), we exploit the following clues describing social relationships between users:

- A user is on the contact list of another user. In this case, it is likely that both users share similar interests.

- A user has written comments on another user's photos.

- A user has saved photos from another user as his favorites.

- Two users belong to the same group.

[2]For YouTube a star-rating is available for the videos but not for other items such as users, groups and tags/concepts.

These relationships can be formalized as social network graphs where the set of vertexes is formed by the users in U:

- *Contact graph* $G_{contact}(U,E)$ with $(u_1,u_2) \in E$ iff user u_2 is in the contact list of user u_1.

- *Comment graph* $G_{comment}(U,E)$ with $(u_1,u_2) \in E$ iff user u_1 has written a comment on a photo of user u_2.

- *Favorites graph* $G_{favorites}(U,E)$ with $(u_1,u_2) \in E$ iff user u_1 has assigned a photo of user u_2 as favorite.

- *Group graph* $G_{group}(U,E)$ with $(u_1,u_2) \in E$ iff user u_1 and user u_2 are are members of the same group.

We can find related users by traversing the social network graphs. For a user u we can, e.g., consider all users that are connected by a path of length $\leq k$, where k is parameter to be determined. In our experiments, we consider only directly connected users in these graphs and compute the utility values analogously as for resources.

Possible extensions are weighted graphs, taking e.g. the number of comments or favorites in $G_{comment}$ or $G_{favorites}$ into account or normalizing the weights in $G_{contact}$ by the overall number of contacts. Furthermore, we can consider combined graphs, computing, e.g. the union of edge-sets of distinct graphs.

Similar relevance clues as described for resources and users can be established for other items such as groups or tags. It should be noted that in the described way, we obtain just relevance values for a subset of items already known to the respective users. In the subsequent paragraphs, we will show how we can extrapolate this and other information to recommend new items to the user.

4.3 Methods for content-driven relationships

For content-based methods, the user will be recommended items similar to those preferred in the past. The simplest, and most direct approach, is to estimate the utility $ut(u,s)$ of item s for user u based on the utilities $ut(u,s_i)$ assigned by user u to items s_i that are 'similar' to s. Formally, given a content representation $Content(s)$ and a content-based profile $ContentBasedProfile(u)$ of a user u, the utility function is usually defined as:

$$ut(u,s) = score(ContentBasedProfile(u), Content(s)) \tag{9}$$

where the *score* function should produce high relevance values if $ContentBasedProfile(u)$ is related to $Content(s)$. We use the vector representations described in Section 3.2 for users and items. Given a vector representation $\vec{u}$ of $ContentBasedProfile(u)$ and $\vec{s}$ of $Content(s)$, the cosine measure can be used as a *scoring* function (or similarity measure) to obtain:

$$ut(u,s) = cos(\vec{u},\vec{s}) = \frac{\vec{u} \cdot \vec{s}}{||\vec{u}|| \cdot ||\vec{s}||} \tag{10}$$

Machine Learning Approach. Alternatively, relevance assignment can be stated as a machine learning problem: given a set of items S_{pos} (represented as feature vectors as described above) relevant to the user, and S_{neg} that are not relevant to the user, train a binary classifier (with the two

classes "relevant for the user" and "not relevant for the user") on these instances. Based on the learned model, it is then possible to estimate the relevance of new items. For Flickr, S_{pos} can be obtained using the user annotations (favorites, comments, contacts, etc.) as described in Section 4.2.

For example, linear support vector machines (SVMs) [29] construct a hyperplane $\vec{w} \cdot \vec{x} + b = 0$ separating the set of positive training examples from a set of negative examples with maximum margin δ. For a new previously unseen, item $\vec{d}$, the SVM simply tests whether the item lays on the "positive" side or the "negative" side of the separating hyperplane. In addition, the distances of the test items from the hyperplane can be interpreted as classification confidences. Alternatively, the relevance estimation can be tackled as a problem of so-called rank learning [5] which aims to automatically learn a function from training samples, such that the function can sort objects (e.g., multimedia resources) according to their degrees of relevance, preference, or importance as defined in a specific user context.

4.4 Methods for social relationships

In *collaborative recommender systems*, also coined *collaborative filtering systems*, the user is recommended items that people with similar preferences have liked in the past. Formally, the utility $ut(u, s)$ of item s and user u is estimated based on the utilities $ut(u_j, s)$ assigned to item s by those users $u_j \in U$ who are similar to user u. The value of an unknown rating $ut(u, s)$ is usually computed as an aggregate of the ratings of other users (e.g. the N most similar) for item s:

$$ut(u, s) = aggr_{u' \in U'} ut(u', s) \quad (11)$$

where U' is the set of the N users most similar to u. Examples for aggregations given in [1] are averaged sum or weighted sum (weighted by the user similarities). Using a similarity measure such as the cosine measure for pairs of users, we can compute the N most similar users. The relevance assignment $ut(u', s)$ can be obtained using implicit ratings of other users described in Section 4.2.

5. EVALUATION

In the previous sections, we have proposed methods for representing objects in folksonomies, using annotations and implicit information, and recommender design. Evaluating recommendations in Web 2.0 applications is a difficult task for several reasons. First, the absence of established reference datasets with large amounts of manually verified and labeled recommendations may require comprehensive user studies with relevance feedback. This makes reliable and reproducible large-scale evaluation very hard and time-consuming. Secondly, there is a significant challenge in deciding what combination of measures should better characterize the recommender quality in a comparative evaluation. Ideally, the evaluation should be objective in reflecting the quality of recommendations with respect to realistic user needs, i.e. capturing the user satisfaction, and be orthogonal to the functionality of the underlying method.

In order to obtain reliable and reproducible results, we are primarily interested in large-scale systematic evaluations with reproducible reference data collections. For this reason, we aim to avoid manual inspection and relevance assignments by a human user. However, the automatic verification of assigned relevance scores is a non-trivial enterprize. In particular, some sources of relevance estimation (e.g. clickthrough data) are usually not publicly available. Therefore, we aim to utilize social aspects of the environment (e.g. favorite lists) for estimating the recommender accuracy. We assume that the ability of the recommender algorithm to reproduce individual user preferences (i.e. favorite lists of preferred resources, contact lists of preferred users, etc.) in automatically generated recommendations reflects the degree of user satisfaction by particular recommendation techniques.

In general, we can expect that explicit user preferences are available only for a small fraction of potentially relevant items. The relevance of further resource, contact, or tag suggestions (which also might be highly suitable for the user) remains open. For this reason, our approach estimates a *lower bound* of the recommender precision, and can be considered as a first step for systematic comparative studies.

users	tags	resources	tag assignments
3,074,947	5,556,568	41,278,715	187,168,654

Table 2: Statistics of the core Flickr data set

contacts	favorites	groups
29,842,973	50,058,103	132,816
group memberships	comments	notes
13,243,481	76,668,998	3,046,794

Table 3: Statistics of the additional information in the Flickr data set

5.1 Data

Our large-scale reference data set was obtained by systematically crawling the Flickr portal during 2006 and 2007. The target of the crawling activity were the core elements of a folksonomy: the users, tags, resources and tag assignments. We also gathered additional information about the interests of the users. The additional information included the contact list of the users, their comments to photos, their favorite photos and memberships in user groups. The size of the crawled data set is summarized in tables 2 and 3.

For crawling the Flickr data set, we applied the following crawling strategy. First, we started a tag centric crawl of all photos that were uploaded between January 2004 and December 2005 and that were still present in Flickr as of June 2007. For this purpose, we initialized a list of known tags with the tag assignments of a random set of photos uploaded in 2004 and 2005. After that, for every known tag we started crawling all photos uploaded between January 2004 and December 2005 and further updated the list of known tags. We stopped the process after we reached the end of the list. After this first part of the crawl we had information about 319,686 users and 28 million photos.

In a second crawl, we started a user centric crawl of Flickr in which we downloaded the public contact lists of all the users known from the first crawl and of all additional users that were found on one of the crawled contact lists. After crawling the contact lists of 3 million users, no additional users were discovered and the crawling came to an end. Beside the contact lists, we crawled for all the users their memberships in user groups and their favorite photos.

In a third crawl, we further extended the set of crawled photos with the information of all photos that were marked

by at least one previously crawled user as one of his favorite photos. This resulted in an overall data set of 41 million photos along with their tag assignment data and comments attached by users.

5.2 Quality measures

A straightforward adaptation of IR-style quality measures is the apriori method with an (estimated) gold standard. Metrics such as precision can be constructed by predicting the k items for which the relevance (or irrelevance) is known. A suitable approximation is achieved by using individual favorite lists (for photos) and contacts (for users), which can be considered as an indication of utility/relevance.

We exploit two possibilities of testing methods for Flickr recommendations:

- The recommender method is constructed in such a way that these dimensions (links to contacts and favorites) remain 'invisible' for the recommendation model.
- An alternative is to keep these dimensions for a training set and evaluate the recommender system on a disjoint test set.

We consider the ability of a recommender method to reconstruct favorite/contact lists as a quality indicator. More precisely, we define the precision of photo/user recommendations as the fraction of recognized favorites/contacts among the top-k recommended items.

5.3 Experimental Design and Results

We considered two characteristic recommendation scenarios discussed above: recommendation of resources (photos) and users. The recommender algorithm was required to produce a ranked list of suggestions for corresponding relations of type user-resource and user-user. The aim of this experiment was to exploit the social dimensions in the Flickr framework and to validate the recommender ability for reconstructing missing social relationships. For this purpose, we selected a set of randomly chosen 1000 "active" users from the Flickr dataset with the following properties: owning at least 50 own resources (photos), having defined at least 90 favorites (explicitly marked distinguished references to resources), having written at least 90 comments (explicitly posted advanced annotations) on other's photos, having defined at least 30 contacts in their contact list, and having not more than 500 photos / favorites / comments / contacts (in order to eliminate less meaningful relationships automatically generated by spammers).

Photo recommendations. For the photo recommender scenario, the favorites of each user were partitioned into disjoint test and "training" subsets (40 training and 50 test favorites per user). Additionally, for each user we randomly selected 250 "contrast" photos not contained in the user's own photos, commented photos, or favorites (in other words, the ratio "positive" test samples vs. "contrast" samples was 1:5). Following our argumentation at the beginning of this section, we considered "contrast" photos as negative test samples in order to estimate the *lower bound* of the recommender precision. Note that choosing a too high number of contrast photos (say several millions of them) would result in a too high probability of having too many relevant photos amongst those; this would distort our experimental strategy.

User representation	Training:10 prec@10	Training:10 prec@20
Random	0.167	0.167
Commented items	0.292	0.280
Favorites	0.757	0.643
	Training:20 prec@10	Training:20 prec@20
Random	0.167	0.167
Commented items	0.296	0.279
Favorites	0.806	0.713
	Training:40 prec@10	Training:40 prec@20
Random	0.167	0.167
Commented items	0.290	0.278
Favorites	0.840	0.757
Personal items	0.254	0.233

Table 4: Photo recommendation scenario for Flickr using global LDA models (average over 1000 users)

For constructing feature vectors of resources, we considered two different latent topics models (Section 3.3). In the first experimental series, we used the 'global' LDA model with 2000 latent topics which was constructed for the entire Flickr dataset, using tagging summaries of particular users as training samples. In the second series, a personalized small-scale LDA model with 16 latent topics was constructed for each user individually. A preliminary pilot study on a small subset of the data showed good results for these numbers of latent topics. In both cases, for user characterization we used tags obtained along the following relationships in the Flickr framework as LDA inputs:

1. **Personal items**: relationships between tags and own user resources
2. **Favorites**: relationships between tags and user favorites (using 10/20/40 favorites for training)
3. **Commented items**: relationships between tags and user comments (using randomly chosen 10/20/40 comments for "training")

After model learning and transformation of user profiles into topical representation, we computed a ranked list of the test vectors most similar to the user profile vector. We considered the reconstruction as successful if the corresponding favorite was observed in the top-10 / top-20 answers within ranked list, and our quality measure was the percentage of the reconstructed favorites among the top items in the result list (prec@10, prec@20). We note that in the presented experimental setting selecting favorite candidates at random would result in a calculational precision of 0.167.

Tables 4 and 5 summarize the results of the photo recommendation scenario. It can be observed that all models provide useful decisions for constructing photo recommenders (with precision clearly greater than random). However, it can be also observed that the precision of "purely" favorite-based recommenders is substantially higher than of the comment- or resource-based ones. This observation reflects the fact that user postings and topics of interest are in general *not* mutually describing. In other words, users that offer photos about holidays and weddings may be primarily

User representation	Training:10 prec@10	Training:10 prec@20
Random	0.167	0.167
Commented items	0.248	0.239
Favorites	0.863	0.719
	Training:20 prec@10	Training:20 prec@20
Random	0.167	0.167
Commented items	0.269	0.252
Favorites	0.878	0.798
	Training:40 prec@10	Training:40 prec@20
Random	0.167	0.167
Commented items	0.273	0.270
Favorites	0.929	0.895
Personal items	0.233	0.228

Table 5: Photo recommendation scenario for Flickr using personal LDA model (average over 1000 users)

User representation	prec@5	prec@10
Random	0.167	0.167
Personal items	0.217	0.214
Commented items	0.364	0.252
Favorites	0.336	0.217
Contacts	0.324	0.219

Table 6: User recommendation scenario for Flickr using global LDA model (average over 1000 users)

User representation	prec@5	prec@10
Random	0.167	0.167
Personal items	0.511	0.414
Commented items	0.582	0.478
Favorites	0.522	0.403
Contacts	0.590	0.463

Table 7: User recommendation scenario for Flickr using personal LDA model (average over 1000 users)

interested in finding images from completely different topics (say Formula 1 or desert hiking) which are not reflected by their own postings at all. Furthermore, it can be observed that a combinational model (i.e., user items, commented items, and favorites are used simultaneously) adds no new value and no further improvement to predictors that are based on favorite-related features only.

User recommendations. For the user recommender scenario, contacts of all 1000 users in the evaluation set were partitioned into disjoint test and "training" subsets (20 training and 10 test contacts per user). For each user we randomly selected 50 "contrast" users not contained in the user's contact list (ratio: "positive" test samples vs. "contrast" samples: 1:5). These test users were represented based on their own resources as described in 3.

The feature vectors for particular user resources were used to construct the aggregated user-specific resource vector (i.e. centroid). The similarity between users was estimated using the cosine similarity measure. For constructing tag-based profile vectors for the 1000 users in the evaluation set, we used tags obtained along the following relationships in the Flickr framework as LDA inputs:

1. **Personal items**: relationships between tags and own user resources
2. **Favorites**: relationships between user's training favorites and resources (using 40 favorites for training)
3. **Commented items**: relationships between tags and user comments (using randomly chosen 40 comments for "training")
4. **Contacts**: users contacts ("training" on 20 contacts)

Analogously to the first evaluation series, we computed a ranked list of the test contacts most similar to the user-specific profile vector. The precision of recommendation was measured as the percentage of reconstructed contacts within the top-5 and top-10 of ranked result set (prec@5, prec@10). Identically to the first experimental series, random ordering of returned recommendations would result in a total precision of 0.167.

Tables 6 and 7 summarize the results of the user recommendation scenario. The results are similar to the scenario of content recommendation. Basically, all models provide useful decisions for constructing contact recommenders (with precision clearly greater than random). However, the use of personal topical models leads to substantially higher recommendation accuracy. Furthermore, in the personal model scenario, the accuracy of contact-based recommenders is consistently higher than of the comment- resource- or favorite-based ones.

Interpretation. The observations from our experimental series emphasize the crucial importance of social relationships and personalization for proper modeling of the user context in a content sharing framework. Until now, the mainstream research was primarily focused on common data dimensions (users, tags, resources) and tripartite relationships between them in a global setting. However, these relationships are of less use for representing and mining meaningful associations in a content sharing environment. Personalized models that capture the user context (using his personal data and the local neighborhood for modeling) often provide higher accuracy at the significantly lower computational and modeling overhead.

6. CONCLUSION AND FUTURE WORK

In this article, we have discussed a design methodology for recommender systems in Web 2.0 folksonomies and we have demonstrated that existing recommender techniques can be carried forward to this new application scenario. The core representational model of our methodology captures dependencies between users, items, annotations, and social aspects (e.g. contacts and favorites) in form of an IR-like vector space model.

The evaluation results with a large-scale Flickr dataset clearly show that the common relationship model between users, resources, and annotations is often not sufficient for constructing accurate recommendation algorithms in folksonomies. The results emphasize the importance of social

aspects, such as contacts or favorites, that have, until now, been rather neglected in recommender scenarios.

Our long-term objective is the design of scalable and reliable assistance methods that individually guide particular users through large-scale Web 2.0 frameworks towards promising search results. In the future, we will conduct the use of multi-modal representational models for integrating low-level object features (e.g. image descriptors) and high-level properties (e.g. tag annotations), as well as combinations of the vector space model with graph-based authority ranking algorithms based on relationships from the user's community-specific social context.

Acknowledgements

This work was supported by EU FP7 projects LivingKnowledge and WeKnowIt and by DFG (German Research Foundation) funded project Multipla.

7. REFERENCES

[1] G. Adomavicius and A. Tuzhilin. Towards the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. Knowl. Data Eng.*, 17(6):734–749, 2005.

[2] R. Albert and A. Barabasi. Statistical mechanics of complex networks. *Review of Modern Physics*, 74:47–97, 2001.

[3] D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[4] R. Burke. Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4):331–370, 2002.

[5] K. Crammer and Y. Singer. Pranking with ranking. *Advances in Neural Information Processing Systems*, 14(1):641–647, 2002.

[6] S. Deerwester, S. Dumais, T. Landauer, G. Furnas, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391Ű407, 1990.

[7] M. Dubinko, R. Kumar, J. Magnani, J. Novak, P. Raghavan, and A. Tomkins. Visualizing tags over time. In *Proc. 15th Int. WWW Conference*, May 2006.

[8] G. Furnas, T. Landauer, L. Gomez, and S. Dumais. The vocabulary problem in human-system communication. *Communications of the ACM*, 30(1):964–971, 1987.

[9] B. Ganter and R. Wille. *Formal Concept Analysis: Mathematical foundations*. Springer, 1999.

[10] T. Hammond, T. Hannay, B. Lund, and J. Scott. Social Bookmarking Tools (I): A General Review. *D-Lib Magazine*, 11(4), April 2005.

[11] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. Riedl. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.*, 22(1):5–53, 2004.

[12] T. Hofmann. Probabilistic latent semantic indexing. *22nd Annual International ACM SIGIR Conference, Berkeley, USA*, pages 50–57, 1999.

[13] A. Hotho, R. Jäschke, C. Schmitz, and G. Stumme. Information Retrieval in Folksonomies: Search and Ranking. In *The Semantic Web: Research and Applications*, volume 4011 of *LNAI*, pages 411–426, Heidelberg, 2006. Springer.

[14] J. Kleinberg. Temporal dynamics of on-line information streams. In M. Garofalakis, J. Gehrke, and R. Rastogi, editors, *Data Stream Management: Processing High-Speed Data Streams*. Springer, 2006.

[15] J. A. Konstan, J. Riedl, A. Borchers, and J. Herlocker. Recommender systems: A grouplens perspective. In *Recommender Systems: Papers from the 1998 Workshop (AAAI Technical Report WS-98-08)*, pages 60–64. AAAI Press, 1998.

[16] R. Lambiotte and M. Ausloos. Collaborative tagging as a tripartite network. ArXiv Computer Science e-prints, Dec. 2005.

[17] F. Lehmann and R. Wille. A triadic approach to formal concept analysis. In *Conceptual Structures: Applications, Implementation and Theory*, volume 954 of *Lecture Notes in Computer Science*. Springer, 1995.

[18] B. Lund, T. Hammond, M. Flack, and T. Hannay. Social Bookmarking Tools (II): A Case Study - Connotea. *D-Lib Magazine*, 11(4), 2005.

[19] S. Middleton, N. Shadbolt, and D. De Roure. Ontological user profiling in recommender systems. *ACM Trans. Inf. Syst.*, 22(1):54–88, 2004.

[20] P. Mika. Ontologies are us: A unified model of social networks and semantics. *4th International Semantic Web Conference (ISWC), Galway, Ireland*, pages 522–536, 2005.

[21] M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45:167, 2003.

[22] K. Nigam, J. Lafferty, and A. McCallum. Using maximum entropy for text classification. *Workshop on Machine Learning for Information Filtering, IJCAI'99*, pages 61–67, 1999.

[23] R. Pastor-Satorras and A. Vespignani. *Evolution and Structure of the Internet: A Statistical Physics Approach*. Cambridge University Press, New York, NY, USA, 2004.

[24] B. J. Schafer, J. A. Konstan, and J. Riedl. Recommender systems in e-commerce. In *ACM Conference on Electronic Commerce*, pages 158–166, 1999.

[25] C. Schmitz, A. Hotho, R. Jaeschke, and G. Stumme. Mining Association Rules in Folksonomies. pages 261–270, 2006.

[26] G. Stumme. A finite state model for on-line analytical processing in triadic contexts. In *ICFCA*, pages 315–328, 2005.

[27] J. Sun, D. Tao, and C. Faloutsos. Beyond streams and graphs: dynamic tensor analysis. *ACM SIGKDD), Philadelphia,USA*, pages 374–383, 2006.

[28] S. van Dongen. A cluster algorithm for graphs. *National Research Institute for Mathematics and Computer Science in the Netherlands, Amsterdam, Technical Report INS-R0010*, 2000.

[29] V. Vapnik. *Statistical Learning Theory*. Wiley, New York, 1998.

[30] R. Wille. Restructuring lattice theory: An approach based on hierarchies of concepts. In I. Rival, editor, *Ordered Sets*, pages 445–470. Reidel, Dordrecht-Boston, 1982.