JIS

Which photo groups should I choose? A comparative study of recommendation algorithms in Flickr

Nan Zheng[1]; Qiudan Li[1]; Shengcai Liao and Leiming Zhang

Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, P.R.China

Abstract

Over the last few years, the social media site Flickr has gained massive popularity. Besides traditional operations on photo sharing, Flickr also offers millions of groups for users to join in order to share photos with others, and the number of groups still increases every day. Choosing among so many options is challenging for users. As such, helping users easily find their desirable groups has become increasingly important. In this paper, we provide a systematic experimental evaluation of several variants of collaborative filtering algorithms to recommend groups to Flickr users. In particular, we develop and compare seven algorithms: three memory-based approaches, and four model-based approaches. Our results suggest that model-based approaches are beneficial compared with memory-based approaches in terms of top-*k* recommendation metric. Specially, models with tags perform well for sparse data, whereas models without tags are more suitable with dense data. Furthermore, incorporating tags in the recommendation algorithms leads to an improvement of precision on the top 2% performance.

Keywords: Flickr Group; Collaborative Filtering; Recommender Systems; Social Tagging Systems.

1. Introduction

With the dramatic development of the Internet, Web 2.0 has emerged and become popular, which transforms users from passive consumers to active producers of content. The social tagging systems, such as Delicious [1], Flickr [2], and CiteULike [3], as typical representatives of Web 2.0, have become increasingly prevalent during the last few years. Social tagging systems allow users to annotate resources with personal labels, called *tags* in order to facilitate later retrieval. Also, users are allowed to store and organize tagging information online through a web interface, instead of storing and organizing them only on the terminal computer. Under such an information management way, the stored tagging information is sharable among users, which makes collaboration available. For instance, people may search for resources annotated by others with a given tag, or browse others' tags for a

[1] *Correspondence to:* Nan Zheng, Qiudan Li, the key Laboratory of Complex Systems and Intelligence Science, Institute of Automation, Chinese Academy of Sciences, No. 95 Zhong Guan Cun East Road, Haidian District, Beijing, P.R.China, 100190, Email: {nan.zheng, qiudan.li}@ia.ac.cn.

given resource. The accessible tagging information may facilitate possible further applications. One example of social tagging systems is Flickr, a photo sharing social media site. As of October 2009, it claims to host more than 4 billion images [4]. Users in Flickr are allowed to update photos, annotate them with tags and share them with appropriate groups. Groups in Flickr are self-organized and explicit definite communities where most groups are related to special topics, such as *portrait*, *animal*, *architecture*, etc. Figure 1 shows a typical part of webpage for a certain group, from which we can find *users* and associated *tags* to be useful information in groups. Negoescu and Gatica-Perez pointed out in [5] that more than half users in Flickr participated in at least one group in their snapshot, which indicated that a large number of users engaged in group activities. The purpose of participating groups is to share photos with others, comment on others' photos and discuss related topics with people of common interests. Thus, the joined groups of a user can reflect the interests of the user. Tags are good indications of users' preferences which have been proved in previous work [6, 7] and tags can also express the topic of the corresponding group. For example, by looking at the tags assigned with a given photo in Figure 1, we may distinguish the topic of the corresponding group to be related to *the sea*, *the sun* and *the clouds*. However, the explosive growth of the group number makes Flickr users become increasingly difficult to find relevant groups they are really interested in. This is the so-called problem of *information overload* [8]. Manually browsing or searching the huge number of groups is very time-consuming and difficult. Thus, it is important to make use of the existing information to discover user's preferences and recommend appropriate groups in order to help users find the desirable groups more easily.

Fig. 1. An example of a certain group and tags assigned to a certain photo.

Recommender systems attempt to help people deal with the information overload problem by filtering huge amounts of information according to users' taste [9]. One of the most successful recommendation technologies is collaborative filtering (CF). CF offers a feasible way by using similar users' past behaviors to generate personalized recommendation. Therefore, it has been broadly used in online applications related to a wide range of areas, such as *movies*, *books*, *music*, *restaurants* and *humor*. When deciding which recommendation algorithm to use, two fundamental factors need to be considered: the quality of recommendation and the algorithm's scalability [10]. Users need recommendations they can trust to help them find resources they will like [11]. If a user finds out he does not like the recommended resources, he may not use the recommender system any more. On the other hand, the ever-growing user-generated contents on the Web bring challenges for real-time recommendation. Algorithms must improve the scalability to meet the time efficiency need of users. However, in most cases, the two factors are in conflict, thus, it is important to make a tradeoff between the quality of recommendation and the scalability.

To alleviate the excessive information overload imposed on Flickr users and to facilitate group participation, we focus on providing group recommendation to assist users to find suitable groups automatically. Most existing works mainly focus on proposing methods for item recommendation in social tagging systems [12-15]. Systematically investigation of the performances of different recommendation algorithms in social tagging systems is rarely explored. In this paper, we develop and compare seven existing and state-of-the-art variants of CF algorithms to recommend groups to Flickr users. Specially, we aim to answer the following research questions:

1. What is the quality of recommendation comparison of applying different approaches to Flickr group recommendation?
2. In which scenarios does each approach manifest its strengths or weaknesses?
3. What is the computational efficiency of each recommendation algorithm?

In this paper, we answer these questions empirically by conducting an empirical study on a real-world dataset. To answer the first question, we compare the top-*k* recommendation performances of the seven models over the entire dataset. To answer the second question, we divide users into two different types and compare the performance differences of the seven models on each type of data. To answer the third question, we empirically compare the time-consuming differences of the seven models. We hope our work could draw more researchers' attention to Flickr group domain, as well as give some preliminary suggestions for the performances of different recommendation algorithms in social tagging systems.

The remainder of this paper is organized as follows: In Section 2, we briefly introduce some related work. Section 3 highlights seven group recommendation models. The experimental evaluations are illustrated in Section 4. Finally, we offer conclusions and possible future directions of our research work in section 5.

2. Literature review

In this section, we briefly present some research literature related to social tagging systems, Flickr group, community recommendations, collaborative filtering and recommendation algorithms in social tagging systems.

2.1. Social tagging sytems

One of the most significant study of social tagging systems appears in the work of Golder and Huberman [16]. The authors took Delicious as an example to analyze the structure and usage pattern of social tagging systems, as well as compared the difference between collaborative tagging and taxonomies. Marlow et al. [17] studied the social aspect of tagging behaviors and showed that the dynamics of interaction and participation in Flickr were different from Delicious. The network structure of folksonomies was investigated in [18], which had been shown similar characteristics to a small-world network. In [19], Sinclair and Cardew-Hall examined whether the tag cloud could provide value to people seeking information from social tagging systems, and the experimental results showed that the tag cloud had its advantages when the information-seeking task was general, but it was not sufficient when the information-seeking task required special information. Their work indicated the importance and necessary to recommend information in social tagging systems to users according to their own and special interests.

2.2. Flickr groups

In particular, we focus on the photo sharing tagging system – Flickr, especially Flickr groups in this article. Flickr groups have begun to be studied in recent literature [5, 20, 21]. The work of [20] and [21] investigated the relationship between photo's popularity and its corresponding group numbers, which indicated that group was one of the main reasons for photo diffusion. In order to understand the dynamics of groups, Negoescu and Gatica-Perez investigated group behaviors in Flickr [5]. Their study showed that the volume of shared photos varied quite a lot among users and the overall group loyalty was quite low. They further analyzed the relationship between user tags and group tags, and proposed a topic-based representation for users and groups based on the usage of tags [22]. In [23], the authors investigated two-mode relations on group-user and group-tag to enrich traditional group search by bringing together semantically similar groups. There is another line of research that focuses on recommending groups for a given photo in Flickr. In [24], the authors proposed a system called *SheepDog*, which suggested groups and tags to an uploaded photo automatically, by considering visual features extracted from the corresponding photo. In [25], photos were suggested to groups based on the image content and user annotations. Three kinds of features were presented in [26] to characterize photos: image content, tags and user communication activity. Then a group recommendation was built by learning a latent space for groups in order to suggest groups to photos. Different from previous work, our research focuses on recommending groups to each user rather than recommending groups to each photo according to photo content.

2.3. Community recommendations

Since Flickr group is a kind of community, our work is also related to community recommendation. Orkut [27] is a social networking website which is affiliated with Google. Like groups in Flickr, users in Orkut may create and join communities to share interests or hobbies. Several work has been conducted in community recommendation in Orkut. For instance, Spertus et al. [28] evaluated different similarity measures to recommend communities to users based on their current community membership. Their recommendation was in a per-community manner, that is, all members of a given community would be offered the same recommendations. On the other hand, Chen et al. [29] proposed a combinational collaborative filtering model for personalized community recommendations in Orkut based on both bags of words and bags of users information. In [30], the authors compared association rule mining (ARM) and latent Dirichlet allocation (LDA) for the community recommendation task in Orkut. The experimental results showed that LDA performed consistently better than ARM for recommending a relatively large number of communities, while ARM was better for recommendation lists of up to 3 communities. In this paper, we compare several CF models to examine their different performances in Flickr group domain.

2.4. Collaborative filtering

Traditional CF uses similar users' past behaviors to generate personalized recommendation. The techniques of CF can be categorized into two classes: memory-based approaches and model-based approaches. Memory-based approaches, including user-based and item-based algorithms [31], predict ratings based on the entire database of previous ratings. They have achieved widespread success in real-life applications for their high efficiency. However, the algorithms have shown reduced coverage and accuracy when the data is sparse [32]. On the other hand, model-based approaches use previous ratings to learn a model, which is then used to produce recommendations. Matrix factorization approaches, such as Singular Vector Decomposition (SVD) [32] and Non-negative Matrix Factorization (NMF) [33], have been proved useful in model-based CF, which predict unobserved user-item pairs by the low-rank factors learned from the observed data in the user-item matrix.

2.5. Recommendation algorithms in social tagging systems

Nowadays, with the dramatic adoption of social tagging systems, a number of studies have focused on recommending items in social tagging systems. Bogers and Bosch [13] applied traditional user-based and item-based CF in CiteULike for recommending scientific articles to users, and found that user-based CF performed better than item-based CF in CiteULike. In [12], a tag-aware fusion recommendation algorithm was proposed to integrate tags in the recommendation process by first extending the user-item matrix and then fusing user-based and item-based CF to produce item recommendation. Recently, tensor decompositions have been applied to solve recommendation problems in social tagging systems. The work of [14] showed how Higher-Order Singular Value Decomposition (HOSVD) could be applied to recommend posts to users in blogging. By extending the work of [14], Symeonidis et al. presented a unified framework for providing recommendations in social tagging systems based on HOSVD, which could recommend items, tags and users simultaneously [34]. Their work indicates that tensor model is effective in social tagging systems. In joining this stream of research, we have proposed an Non-negative CANDECOMP/ PARAFAC (NNCP) decomposition-based recommendation model to discover latent associations among the three entities in social tagging systems and demonstrated its effectiveness [15]. However, to the best of our knowledge, there is little systematic research on comparing the performances of different recommendation algorithms in social tagging systems. In this paper, we investigate a comparison study of different item recommendation algorithms in Flickr group domain to analyze their qualities of recommendation and the efficiencies.

3. Recommendation algorithms

We now present seven types of representative CF algorithms including three memory-based, two matrix factorization (MF)-based and two tensor decompositions (TD)-based approaches. We focus on Flickr group recommendation that is based on users' past group-participation behaviors. The output of an algorithm is considered to be predictions of groups for individual users that represent possibilities of future participations. A ranked list of *k* groups with the highest prediction scores for a target user serves as the recommendations. Table 1 lists some notations used throughout this paper.

Table 1
Notations

N_u	number of users
N_t	number of tags
N_g	number of groups
k	number of requested recommendations
$u \in \{1,\ldots,N_u\}$	index for users
$t \in \{1,\ldots,N_t\}$	index for tags
$g \in \{1,\ldots,N_g\}$	index for groups

3.1. Memory-based Approaches

For each user, the number of photo uploading is counted on each group, and used to form the user-group matrix $A \in \mathrm{R}^{N_u \times N_g}$. Each row of *A* contains the photo counts in different groups for a particular user, whereas each column of *A* contains the photo counts by different users in a particular group.

3.1.1 User-based group recommendation model

The user-based model [11] predicts a target user's future participation in a group as a two-step process. First, the neighborhood of the target user is computed by the similar group-participation pattern between the target user and other users. That is, the similarity is calculated on the row vectors of *A*. We use the commonly used cosine similarity metric [35] to determine similarity between users. Once *b* neighbors have been identified according to the similarity scores, all groups that *b* neighbors have participated but have not been joined by the target user are sorted by their weights and considered to be possible recommendations.

3.1.2 Item-based group recommendation model

The item-based model is proposed in [31] which is different from the user-based model only in that group similarities are calculated instead of user similarities. That is, the similarity is calculated on the column vectors of *A*. We also use cosine similarity metric to determine similarity between groups. A higher similarity score indicates that the two groups have been co-joined by many users. Then, *d* groups that have the high similarity scores with each joined group of the target user are selected and finally, groups are sorted by their weights and recommended to the target user.

3.1.3 Tag-aware group recommendation model

The tag-aware model [12] is a generic model that allows tags to be incorporated to standard memory-based CF algorithms, by reducing the three-dimensional correlations to three two-dimensional correlations. Based on their work, we design a tag-aware group recommendation model. First, user-group matrix is extended with user-tag matrix to form user-group|tag matrix. Similarly, group-user matrix is extended with group-tag matrix to form group-user|tag matrix. The user-based recommendation is applied on the user-group|tag matrix to find user's neighborhood based on both group-participation pattern and tag-usage pattern. The item-based recommendation is carried on the group-user|tag matrix to calculate group similarities by considering both user-participation pattern and tag-assignment pattern. Finally, a linear fusion function with parameter λ is applied to combine user-based and item-based predictions.

3.2. MF-based Approaches

In MF-based approaches, we use the same user-group matrix *A* to conduct group recommendations. The basic idea of MF is to fit matrix *A* with a low rank approximation and use it to make further predictions.

3.2.1 SVD-based group recommendation model

SVD is a classical dimensionality reduction-based algorithm which decomposes the user-group matrix *A* into $U \cdot Z \cdot V^T$, where *U* and *V* are two orthogonal matrices of size $N_u \times R$ and $N_g \times R$ respectively and R is the rank of matrix *A*. *Z* is a diagonal matrix of size R×R with all singular values of matrix *A* as its diagonal entries. Then the matrix *Z* is reduced by retaining only *n* largest singular values to form Z_n, *U* and *V* are reduced accordingly to form U_n and V_n. The prediction score of user *u* to group *g* is calculated based on the product of row vector of $U_n Z_n^{1/2}$ and the column vector of $Z_n^{1/2} V^T$, such that:

$$\hat{A}_{ug} = (U_n Z_n^{1/2})_{u:} (Z_n^{1/2} V^T)_{:g} \qquad (1).$$

$\hat{A}_{ug}$ gives the potential score of group *g* to user *u*, namely the likeliness that user *u* will participate in group *g*. Finally, a list of top-*k* groups is recommended for user *u* according to the potential scores [32].

3.2.2 NMF-based group recommendation model

NMF is a method to fit user-group matrix *A* with a low rank non-negative matrix, such that $\hat{A} = WH$, where $W \in R^{N_u \times T_o}$, $H \in R^{T_o \times N_g}$ with non-negativity constraint, and T_o is the number of the decomposed factors. The goal is to find $\hat{A}$ that minimizes a loss measure, such as sum of the squared distances. By using the non-negativity part of the Kuhn-Tucker condition, we obtain the following update rules [36]:

$$W_{uy} \leftarrow W_{uy} \frac{(AH^T)_{uy}}{(WHH^T)_{uy}} \qquad H_{yg} \leftarrow H_{yg} \frac{(W^T A)_{yg}}{(W^T WH)_{yg}} \qquad (2).$$

The low rank matrix $\hat{A}$ captures latent associations between users and groups through T_o factors that allows us to compute the predicted likeliness that a user will participate in a group. To this end, we make use of $\hat{A}$ to implement Flickr group recommendation. The advantages of NMF are that (1) non-negative values of the result factor matrices facilitate easier interpretation and (2) the non-negative constraints lead to a sparsely distributed representation. See [33] for a detailed description of how NMF can be applied in recommender systems.

3.3. TD-based Approaches

Tensor decompositions are higher-order extensions of matrix factorizations that capture the underlying patterns in multi-mode data. In TD-based approaches, a three-mode tensor $\mathcal{X} \in \mathbb{R}^{N_u \times N_t \times N_g}$ can be constructed from the usage data in Flickr groups. Each element x_{utg} in the tensor denotes user *u*'s corresponding photo count with tag *t* in group *g*. The advantage of a three-way tensor is that we can explicitly model the relations among users, tags and groups in a unified way.

3.3.1 HOSVD-based group recommendation model

Most existing algorithms in social tagging systems split the three dimensional space into pair relations in order to apply already existing techniques, such as CF and link mining. Therefore, they miss a part of the total interaction between the three dimensions. Thus, in [14], Symeonidis and Deligiaouri developed a unified framework to concurrently model the three types of entities. To reveal latent semantics, they performed three-mode analysis using HOSVD. HOSVD decompositions a tensor into a core tensor multiplied by a matrix along each mode. The algorithm first computes the matrix unfoldings[2] of $\mathcal{X}$ to build three new matrices $X_{(1)} \in \mathbb{R}^{N_u \times N_t N_g}$, $X_{(2)} \in \mathbb{R}^{N_t \times N_u N_g}$ and $X_{(3)} \in \mathbb{R}^{N_g \times N_u N_t}$. Then, SVD is applied in each new matrix, such that $X_{(i)} = U^{(i)} \cdot Z^{(i)} \cdot (V^{(i)})^T$ (i=1,2,3). After retaining the left singular vectors from $U^{(i)}$ corresponding with the c_i largest singular values, denoted as $U_{c_i}^{(i)}$, a core tensor $\mathcal{S}$ of size $c_1 \times c_2 \times c_3$ is constructed by the mode products[3] of the three matrices as follows:

$$\mathcal{S} = \mathcal{X} \times_1 (U_{c_1}^{(1)})^T \times_2 (U_{c_2}^{(2)})^T \times_3 (U_{c_3}^{(3)})^T \qquad (3).$$

Finally, the reconstructed tensor $\hat{\mathcal{X}}$ is built by the product of the core tensor $\mathcal{S}$ and the mode products of the three matrices:

$$\hat{\mathcal{X}} = \mathcal{S} \times_1 U_{c_1}^{(1)} \times_2 U_{c_2}^{(2)} \times_3 U_{c_3}^{(3)} \qquad (4)$$

$\hat{\mathcal{X}}$ measures the associations among users, tags and groups, and each element $\hat{x}_{utg}$ of $\hat{\mathcal{X}}$ represents the likeliness that user *u* will upload a photo to group *g* with tag *t*. To this end, groups can be recommended according to the scores associated with {user, tag} pair. HOSVD can be viewed as a natural extension to SVD. See [14] for a detailed description of HOSVD-based recommendation algorithm.

3.3.2 NNCP-based group recommendation model

In our previous research, we have introduced a tensor decomposition-based group recommendation model based on NNCP [15]. The basic idea of NNCP is to fit $\mathcal{Z}$ ($\mathcal{Z} \in \mathbb{R}^{N_1 \times N_2 \times N_3}$) with a sum of non-negative component rank-one tensors [37], thus in the three-mode case:

---

[2] Unfolding is the process of reordering the elements of a tensor into a matrix. Unfolding a tensor $\mathcal{X} \in \mathbb{R}^{N_1 \times \ldots \times N_M}$ on mode *d* results in a matrix $X_{(d)}$ of size $N_d \times N_1 \ldots N_{d-1} N_{d+1} \ldots N_M$.

[3] The *d*-mode product of a tensor $\mathcal{X} \in \mathbb{R}^{N_1 \times \ldots \times N_M}$ with a matrix $U \in \mathbb{R}^{J \times N_d}$ is denoted by $\mathcal{X} \times_d U$. The result is an $(N_1 \times \ldots \times N_{d-1} \times J \times N_{d+1} \times \ldots \times N_M)$-tensor, where $(\mathcal{X} \times_d U)_{n_1 n_2 \ldots n_{d-1} j n_{d+1} \ldots n_M} = \sum_{n_d = 1}^{N_d} x_{n_1 n_2 \ldots n_M} u_{j n_d}$.

$$Z \approx \sum_{r=1}^{T_o} \lambda_r \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r \qquad (5)$$

where the symbol $\circ$ denotes the outer product, λ_r is a weight that indicates how important the r-th component is, and vector $\mathbf{a}_r \in \mathrm{R}^{N_1}$, $\mathbf{b}_r \in \mathbf{R}^{N_2}$ and $\mathbf{c}_r \in \mathbf{R}^{N_3}$ are normalized to length one. The decomposition process is illustrated in Figure 2.

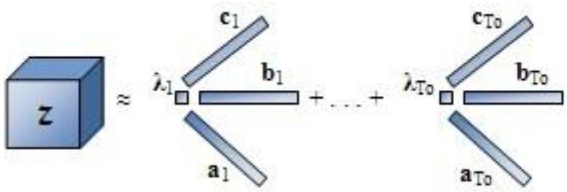

Fig. 2. CP decomposition of a 3-mode tensor.

We introduce NNCP-based group recommendation model in two parts: (1) how to discover the latent topics through the interactions among users, tags and groups, and (2) how to implement group recommendation based on latent topics.

Latent topic extraction

We aim to find the latent topics from three-relational social data represented by a three-mode tensor. Our goal is to discover latent associations from the extracted latent topics. Specially, we are interested in finding the latent associations between users and unjoined groups. For example, a user who has a great number of semantic tag overlaps with a group, or a group which has been joined by users who have similar tastes with the given user.

The latent associations between user u and group g, written as x_{ug}, can be considered to be a function of the relation between topic r and user u, and topic r with group g. From the perspective of probability, let p_{ur} denotes how likely user u is related to the r-th topic, p_{gr} indicates how likely group g is associated with the r-th topic, and p_r is the probability of the r-th topic, then we can express x_{ug} by $x_{ug} \approx \sum_{r=1}^{T_o} p_{ur} \cdot p_{gr} \cdot p_r$. Extending it to three-mode relations, we get $x_{utg} \approx \sum_{r=1}^{T_o} p_r \cdot p_{ur} \cdot p_{tr} \cdot p_{gr}$. Then the set of such interactions can be written by:

$$X \approx \sum_{r=1}^{T_o} p_r \mathbf{u}_r \circ \mathbf{t}_r \circ \mathbf{g}_r \qquad (6)$$

where vector $\mathbf{u}_r \in \mathrm{R}^{N_u}$, $\mathbf{t}_r \in \mathrm{R}^{N_t}$ and $\mathbf{g}_r \in \mathrm{R}^{N_g}$. This is similar to the CP decomposition shown in Equation 5. Combining the vectors from the rank-one components, we get three factor matrices: user matrix U=[$\mathbf{u}_1 \ldots \mathbf{u}_{To}$], tag matrix T=[$\mathbf{t}_1 \ldots \mathbf{t}_{To}$] and group matrix G=[$\mathbf{g}_1 \ldots \mathbf{g}_{To}$]. The goal is to find an approximation that minimizes a loss function such as the following sum of the squared distances:

$$D = \sum_{u=1}^{N_u} \sum_{t=1}^{N_t} \sum_{g=1}^{N_g} [x_{utg} - (\sum_{r=1}^{T_o} p_r \mathbf{u}_r \circ \mathbf{t}_r \circ \mathbf{g}_r)_{utg}]^2 \qquad (7)$$

Since negative values lack a physical meaning, we adopt non-negative constrains on the result factor matrices, which facilitates easier interpretation. In this way, this optimization problem falls into the category of non-negative matrix factorization (NMF) [38]. After initializing U, T and G by random non-negative values, we perform the multiplicative updates, which are like the updates rules for NMF:

$$U_{ur} \leftarrow U_{ur} \frac{(X_{(1)}(G \odot T))_{ur}}{(U(G \odot T)^T (G \odot T))_{ur}} \qquad (8)$$

$$T_{tr} \leftarrow T_{tr} \frac{(X_{(2)}(G \odot U))_{tr}}{(T(G \odot U)^T (G \odot U))_{tr}} \qquad (9)$$

$$G_{gr} \leftarrow G_{gr} \frac{(X_{(3)}(T \odot U))_{gr}}{(G(T \odot U)^T (T \odot U))_{gr}} \qquad (10)$$

where $\odot$ denotes the Khatri-Rao product [39], $X_{(1)}$, $X_{(2)}$ and $X_{(3)}$ are three matrix unfoldings of X, respectively. At each step, one of the matrices *U*, *T*, *G* is updated, while the other two are kept constant. The non-negative initial conditions and multiplicative updates preserve the non-negativity constraints of *U*, *T* and *G*. In this way, NNCP can be considered as a higher dimensional of NMF.

Group prediction

The three factor matrices discover the latent topics that govern the associations among users, tags and groups. The *u*-th row of matrix *U*, denoted with $\mathbf{u}_u \in \mathrm{R}^{\mathrm{R}}$, provides an additive linear combination of factors which indicates the topics of user *u*. The higher weight user *u* is assigned to a factor, the more interests user *u* has in the relevant topic. The *g*-th row of matrix *G* ($\mathbf{g}_g \in \mathrm{R}^{\mathrm{R}}$) provides an additive linear combination of factors which indicates the topics of group *g*. The higher weight a group is assigned to a factor, the more related the group is with the relevant topic. Consequently, groups can be recommended according to the captured associations. We define a score matrix S_c ($N_u \times N_g$) as follows:

$$S_c = \sum_{r=1}^{\mathrm{T_o}} p_r \mathbf{u}_r \mathbf{g}_r^{\mathrm{T}} \qquad (11)$$

A high similarity score $S_c(u,g)$ indicates that user *u* has a great probability to join group *g* in the future. Therefore, for user *u*, groups can be recommended according to the scores.

A full procedure for NNCP-based group recommendation model is shown in Algorithm 1.

Algorithm 1 NNCP-based model

Input:

X – user-tag-group tensor

$\mathrm{T_o}$ – number of topics

k – number of requested recommendations

M – the maximum times of iteration

ε – reconstruction error

Main procedure:

Initialize *U*, *T* and *G* with random non-negative values

Normalize columns of *U*, *T*, *G*, and store norms as *p*

Initialize loss function $D_0 = \left\| \mathrm{X} - \sum_{r=1}^{\mathrm{T_o}} p_r \mathbf{u}_r \circ \mathbf{t}_r \circ \mathbf{g}_r \right\|^2$

for *m*=1 to M **do**

Update *U*, *T* and *G* using Equation 8-10

Normalize columns of *U*, *T*, *G*, and store norms as *p*

Compute loss function $D_m = \left\| \mathrm{X} - \sum_{r=1}^{T_o} p_r \mathbf{u}_r \circ \mathbf{t}_r \circ \mathbf{g}_r \right\|^2$

if $|D_m - D_{m-1}| < \varepsilon$ stop

else go on

end for

Compute score matrix S_c using Equation 11

for u=1 to N_u **do**

Remove the groups that user u has joined in the row vector $S_c(u)$

Sort $S_c(u)$ in the descending order

end for

Output: The first k groups sorted for each user

4. EXPERIMENTS

4.1. Dataset

We use Flickr API [40] to gather data about users, tags and groups. Flickr offers a special API named flickr.photos.getRecent, which returns a list of the latest public photos as well as theirs owners who have uploaded them to Flickr. We use it to download the latest active users on Dec 14th, 2009. For each user, we collect all his joined public groups and all his tags for public photos. After removing the users who have not joined any groups and users who have not annotated photos with any tags, in total, we collect 41,229 groups and 211,735 tags for 620 users. In order to reduce noise and focus on the denser portion of the dataset, we filter out users who have tagged less than 20 photos, tags have turned up less than 20 times and groups which have been joined by less than 10 users. Then, after stop-word removal and stemming, we end up with 336 users, 12,314 tags and 19,425 groups.

Flickr offers special groups to award high quality photos. These photos are invited to special groups, such as "Artistic Expressions - Invited Images Only" and "***Rose Award Group", by other users but not uploaded by the owners. Since it is not the initial willing of the owner to upload photos to the special groups, it does not make any sense if recommending such special groups. Therefore, we remove a list of 1598 special groups from the original dataset. Finally, our dataset consists of 336 users, 12,314 tags, 17,827 groups, and over 7 million {user, tag, group} triples.

4.2. Evaluation Metric and Protocol

We use the top-*k* recommendations metric [41] to evaluate group recommendation results. That is, we suggest the top *k* groups for each user. We randomly select one joined group and *k*-1 unjoined groups for each user to form the test set and the remaining joined groups form the training set. The objective is to find the place of the joined group in the recommendation list. There are *k* possible ranks for the joined group and the best result is that no unjoined groups appear before the joined one in the recommendation list. In order to establish statistical significance of the findings, we have repeated the procedure of training/test partition 10 times with different random seeds. The reported numbers are the mean performance averaged over the 10 runs. In addition, we set *k* to be 1001 in the experiments.

4.3. Parameter Settings

The choice of parameters is critical to high-quality and efficient group recommendation. We perform parameter tuning in advance for each model and then use the best settings in the experiments.

- User-based model: We vary user neighborhood size b from 10 to 100 by an interval of 10, and find the best b to be 40.
- Item-based mode: We vary group neighborhood size d from 10 to 100 by an interval of 10, and find the best d to be 60.
- Tag-aware model: After fixing user neighborhood size b to be 40 and group neighborhood size d to be 60, we examine the impact of λ experimentally. λ=0.8 gives the most optimum results.
- SVD-based model: We repeat the process by preserving 10%, …, 100% (by an interval of 10%) of the original diagonal of A to produce group recommendation. We find 40% to be the most optimum value. A higher percentage will cause over-fitting problem. We then fix n to be the number of singular values by preserving 40% of the original diagonal of Z in each run.
- NMF-based model: The ideal value of T_o should be large enough to capture all the important characterizations yet small enough to gain time efficiency. We range T_o from 10 to 140 by an interval of 10. We find that T_o=80 is basically enough to represent the latent topics on our dataset. As to iteration, 50 times can provide satisfying result. Thus, in our experiments, we set number of topics T_o to be 80 and times of iteration to be 50.
- HOSVD-based model: Since there is no straightforward way to find the optimal values for c_1, c_2 and c_3, we follow the way according to [34] that a 70% of the original diagonal of $X_{(1)}$, $X_{(2)}$ and $X_{(3)}$ matrices can give good approximations. Thus, c_1, c_2 and c_3 are set to be the numbers of singular values by preserving 70% of the original diagonal of $X_{(1)}$, $X_{(2)}$ and $X_{(3)}$ respectively in each run. Specially, we apply the framework of MET [42] for HOSVD to solve the problem of memory overflow. Such a framework can maximum the computation speed while optimally utilizing the available memory.
- NNCP-based model: The parameters in NNCP are in the similar situation with the parameters in NMF. We find that T_o=90 is basically enough to represent the latent topics on our dataset. As to iteration, 50 times can provide satisfying result. Therefore, we set number of topics T_o to be 90 and times of iteration to be 50.

4.4. Results and Discussions

In this subsection, we conduct experimental comparison of the seven CF algorithms when recommending groups for Flickr users. We present the experimental results in three perspectives: (1) top-k recommendation performances of the seven algorithms, (2) different performances of the seven algorithms when applying on different user types, and (3) the computational efficiency of each algorithm.

4.4.1 Top-k recommendation performances

Figure 3 shows the cumulative quantity of ranks for the joined groups in the test set. 0% means that the joined group is at the first place in the ordered list, while 100% means that it turns up at the last position. As shown, model-based approaches achieve better performances than memory-based approaches. It may due to the low coverage of memory-based models under sparse data. We can see that after the top 60% rank, memory-based models seem to have a relatively flat trend, that is, few joined groups are recommended among these positions. However, most joined groups turn up at the last position (100%), which suggests that these joined groups are not included in the neighbors. Note that user-based model performs significantly better than item-based model. This may due to the fact that the user-dimension is much denser than the group-dimension in our dataset.

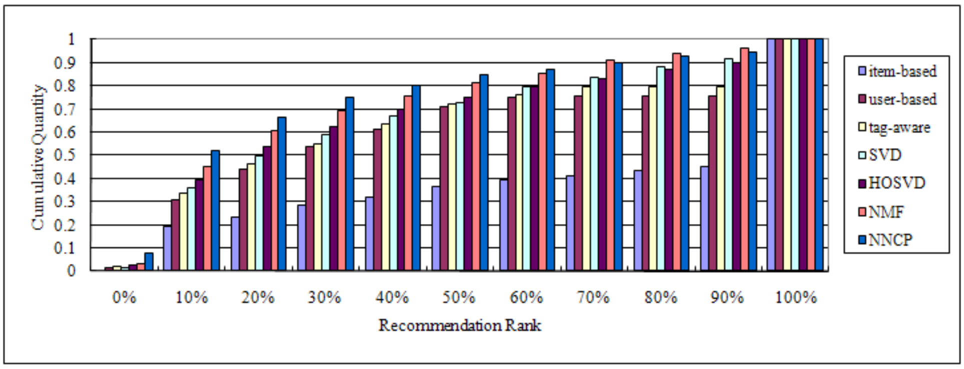

Fig. 3. Top-1001 recommendation performances.

Considering model-based approaches only, both NNCP- and NMF-based models achieve better performances than HOSVD- and SVD-based models, which indicates the overall performance of recommendation algorithm is mainly due to the basic method taken by the algorithm. The non-negative constraints lead to the sparsely distributed representation which makes more sense for each user and each group to be associated with (or concentrated on) some small subset of a large array of topics. Thus, it may capture more accurate associations than SVD method, which leads to the higher performance. Additionally, TD approaches generally achieve better performance than their MF counterparts before the top 60% rank and then MF approaches slightly outperform TD models for the rest of the ranks. In real life, users are typically much more interested in the quality of the top 20 recommendations and the performance of the algorithms at the 100th to 1000th recommendations is practically irrelevant [43]. Thus, we illustrate the cumulative distribution of ranks on the top 2% performance in Figure 4. It can be seen that models with tags (e.g. tag-aware, HOSVD, and NNCP) achieve better results on the top 2% ranks than their counterparts that without tags (e.g. user-based, item-based, SVD and NMF). The good performance seems to indicate that adding tags can improve the quality of recommendation for the top ranks.

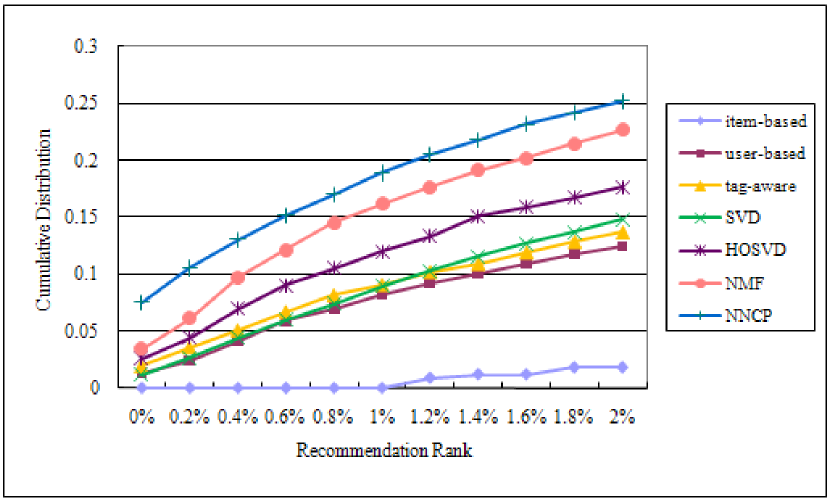

Fig. 4. Top 2% ranks of recommendation performances.

4.4.2 Performances with different user types

In order to compare the seven models more thoroughly, we compare the performance on users with different number of groups. Figure 5 shows the group number of each user.

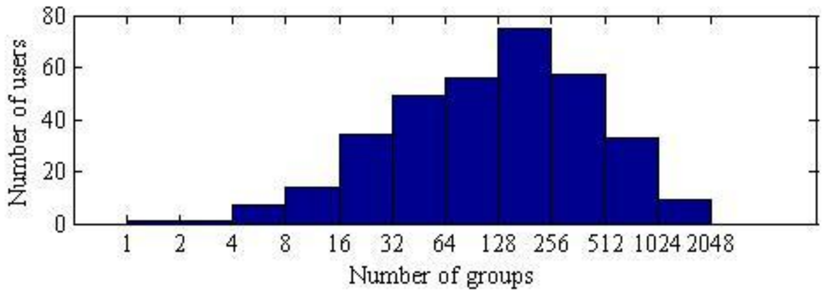

Fig. 5. Histogram of number of groups per user.

As shown, users in Flickr differ widely with respect to how many groups they have participated. We cut the data into two sets, named as *Small* and *Large*, according to the number of groups a user has participated. *Small* refers to the users who have participated in less than 150 groups while *Large* refers to the users with at least 150 groups. Table 2 reports the datasets of each type of users. We perform seven algorithms on each user type, and Figure 6 and Figure 7 display the cumulative quantity of ranks for the joined group.

Table 2

Dataset characteristics

Property	*Small*	*Large*
Number of users	170	166
Number of tags	7988	10974
Number of groups	5014	14146
Average number of groups per user	58.79	541.04
Group overlap rate[4]	49.8%	84.3%

Looking across the same model in each dataset (see Figure 6 and Figure 7), we find that all algorithms achieve better performance with *Large* dataset than *Small* dataset. The difference is even more significant for user-, NMF- and SVD-based models. The good performance with *Large* dataset may be associated with the dataset's much higher average density levels (4.114% for user-group matrix and 0.021% for user-tag-group tensor) compared to *Small* dataset (1.504% for user-group matrix and 0.011% for user-tag-group tensor).

For *Small* dataset, models with tags perform much better than their counterparts without tags (Figure 6). The difference is statistically significant ($p<0.01$). This is perhaps because for *Small* dataset there are fewer group overlaps among users (49.8% as shown in Table 2), and fewer associations exist between users and unobserved groups in two-mode relations. Thus, the coverage of groups available for recommendations becomes smaller. While for the models with tags, semantic tags are involved in mining associations between users and groups, and it addresses sparsity and deals with the *cold-start* problem to some extent.

For *Large* dataset, the performance difference of models with tags and the corresponding models without tags is minor. Models with tags perform slightly better than the corresponding models without tags before 50% and then the performance gaps are very minor (Figure 7). The possible reason is that in *Large* dataset, there are sufficient group overlaps among users (84.2% as shown in Table 2), and in this case, models without tags can work well to capture associations between users and unobserved groups in two–mode relations. However, as more

[4] Group overlap rate is the percentage of the number of groups which have been joined by at least two users to the overall group number.

tags added, more noises (ambiguous tags) may be involved, which causes relatively low performance of models with tags.

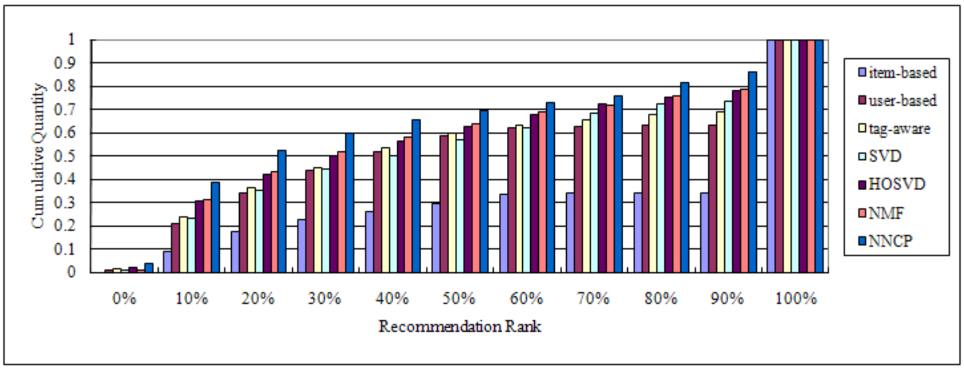

Fig. 6. Top-1001 recommendation performance for *Small*.

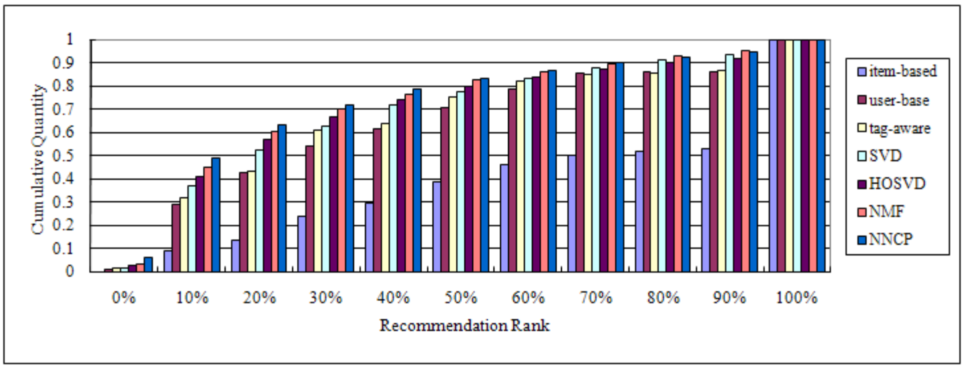

Fig. 7. Top-1001 recommendation performance for *Large*.

Figure 8 illustrates the cumulative distribution of ranks on the top 2% performance under *Small* dataset and *Large* dataset. For both user types, models with tags perform better than their corresponding models without tags for the top 2% ranks. The difference is statistically significant ($p<0.01$). Another interesting observation is that the curves of models with tags grow steadily from 0% to 2%, while the curves of models without tags are much steeper at the beginning, which indicates adding tags can bring more accurate predictions for the top ranks.

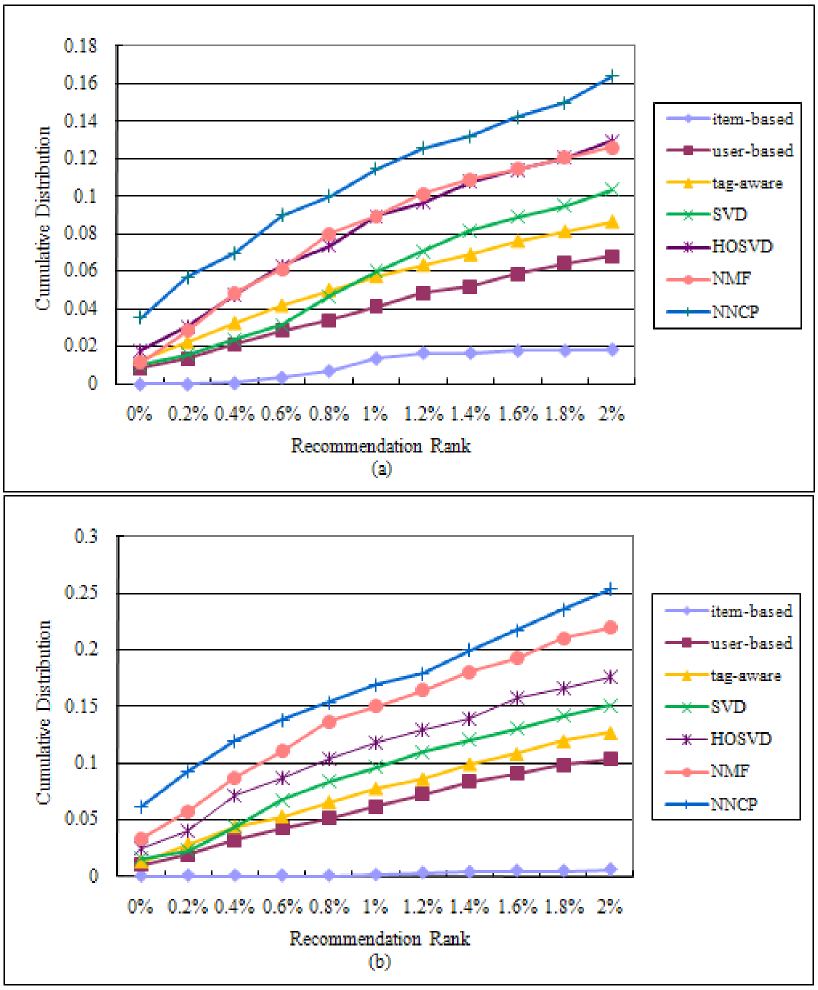

Fig. 8. (a) Top 2% ranks for *Small* dataset, and (b) top 2% ranks for *Large* dataset.

4.4.3 Computational efficiency analysis

The time complexity of the algorithm is critical to real time recommender system. We now report the empirical computational efficiency of the seven recommendation algorithms on the entire dataset. All of our experiments were performed on a 3 GHz Pentium IV, with 16GB of memory, running Windows Server 2003. Figure 9 summarizes total computing times of all algorithms. (All algorithms were implemented in Matlab.) As shown, NNCP requires the longest running time. It is because in each iteration, the algorithm needs to update three matrices as well as compute sum of the squared distances between the original tensor and reconstructed tensor, both of which require significant CPU cycles. Tag-aware is time consuming because of the high dimension of the extended matrices. HOSVD is slow due to the reduction of three high dimensional unfolding matrices. Item-based model is also slow due to the high group-dimension (relative to user-dimension) in the dataset. SVD and NMF require modest computing times. User-based model is especially fast because of the low user-dimension in the dataset as well as the simpleness of the algorithm. It is clear that models with tags are time consuming; however, the computations can be performed offline in advance. For real-world applications, the use of parallel architectures may further reduce execution times.

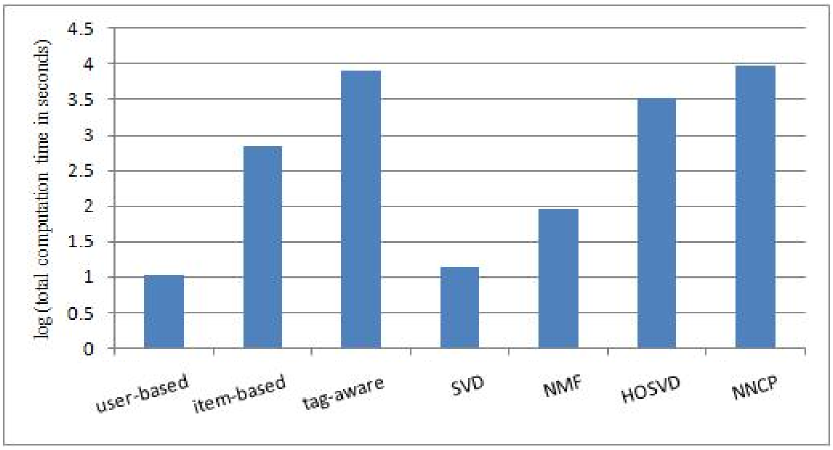

Fig. 9. Computational efficiency analysis.

4.4.4 Quality vs. efficiency

Based on the above analysis, no single algorithm is observed to dominate other algorithms across all experimental configurations, but we can get some insights from the results of Flickr group recommendation. First, when the applications concern high efficiency rather than the quality of recommendation, two-mode memory-based models may be good choices. Specially, when user size is less than item size, user-based CF may gain better performance; when item size is less than user size, item-based CF is more suitable. Also, memory-based methods have the advantage of being able to provide more transparency through simple explanations, comparing to model-based methods. Second, if the applications need high quality of recommendation but could accept the time consuming problem, models with tags are good choices, especially for sparse data. Finally, MF-based model may be a good tradeoff between the quality of recommendation and the scalability for dense data.

5. Conclusions and future work

A unique contribution of this study is a comprehensive evaluation of a wide range of CF algorithms for Flickr group recommendation. Our study can give some preliminary suggestions for the item recommendation applications in social tagging systems. First, model-based approaches generally perform better than memory-based approaches in terms of top-*k* recommendations metric. Second, models with tags outperform the corresponding models without tags for sparse data, while models without tags achieve better performance for dense data and are more efficient than the corresponding models with tags. Third, incorporating tags in the recommendation algorithms can help to obtain more accurate recommendation in the top 2% ranks. For future work, we plan to consider group activity in the process of group recommendation, such as group size, member size and frequency of uploading photos, in order to provide more accurate group recommendation for Flickr users.

6. Acknowledgments

This research was supported in part by the following grants: Program 863 "Key Technologies on Cross-media Search and Service Product Development", Program 973 "Comprehensive Situation Assessment and Decision-Making Technologies Based on Unstructured Information Processing", and NSFC, "Personalized Recommendation Technologies in Mobile Commerce", No. 60703085.

7. References

[1] *Delicious*. Available at: http://delicious.com/ (accessed 14 December 2009).

[2] *Flickr*. Available at: http://www.flickr.com/ (accessed 14 December 2009).

[3] *CiteULike*. Available at: http://www.citeulike.org/ (accessed 14 December 2009).

[4] *Flickr Blog*. Available at: http://blog.flickr.net/en/2009/10/12/4000000000/ (accessed 14 December 2009).

[5] R. Negoescu and D. Gatica-Perez, Analyzing flickr groups. *Proceedings of the 2008 International Conference on Content-based Image and Video Retrieval* (Niagara Falls, Ontario, Canada, July 7-9, 2008). *CIVP'08* (ACM Press, New York) 417-426.

[6] H. Kim, S. Decker and J. Breslin, Representing and sharing folksonomies with semantics. *Journal of Information Science*. 36(1) (2010) 57-72.

[7] A. Shepitsen, J. Gemmell, B. Mobasher and R.D. Burke, Personalized recommendation in social tagging systems using hierarchical clustering. *Proceedings of the 2008 ACM conference on Recommender systems* (Lausanne, Switzerland, October 23-25, 2008). *RecSys'08* (ACM Press, New York) 259-266.

[8] D. Bawden and L. Robinson, The dark side of information: overload, anxiety and other paradoxes and pathologies. *Journal of Information Science*. 35(2) (2009) 180-191.

[9] P. Resnick and H. Varian, Recommender systems. *Communications of the ACM*. 40(3) (1997) 56-58.

[10] J.L. Herlocker, J.A. Konstan, L.G. Terveen and J. T. Riedl, Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*. 22(1) (2004) 5-53.

[11] B. Sarwar, G. Karypis, J. Konstan and J. Riedl, Analysis of recommendation algorithms for e-commerce. *Proceedings of the 2nd ACM conference on Electronic commerce* (Minneapolis, Minnesota, October 17-20, 2000). *EC'00* (ACM Press, New York) 285-295.

[12] K.H.L. Tso-Sutter, L.B. Marinho and L. Schmidt-Thieme, Tag-aware recommender systems by fusion of collaborative filtering algorithms. *Proceedings of the 2008 ACM symposium on Applied computing* (Fortaleza, Ceara, Brazil, March 16-30, 2008). *SAC'08* (ACM Press, New York) 1995-1999.

[13] T. Bogers and A. van den Bosch, Recommending scientific articles using citeulike. *Proceedings of the 2008 ACM conference on Recommender systems* (Lausanne, Switzerland, October 23-25, 2008). *RecSys'08* (ACM Press, New York) 287-290.

[14] P. Symeonidis and A. Deligiaouri, (2010) Recommending Posts in Political Blogs based on Tensor Dimensionality Reduction. *International Journal of Engineering Intelligent Systems* (2010).

[15] N. Zheng, Q. Li, S. Liao and L. Zhang, Flickr group recommendation based on tensor decomposition. *Proceedings of the 33rd annual international ACM SIGIR conference on Research and development in information retrieval ACM* (Geneva, Switzerland, July 19-23, 2010). *SIGIR 2010* (ACM Press, New York).

[16] S. Golder and B.A. Huberman, Usage patterns of collaborative tagging systems. *Journal of Information Science*. 32(2) (2006) 198-208.

[17] C. Marlow, M. Naaman, D. Boyd and M. Davis, HT06, tagging paper, taxonomy, Flickr, academic article, to read. *Proceedings of the 17th conference on Hypertext and hypermedia* (Odense, Denmark, August 22-25, 2006). *HYPERTEXT'06* (ACM Press, New York) 31-40.

[18] C. Cattuto, C. Schmitz, A. Baldassarri, V. Servedio, V. Loreto, A. Hotho, M. Grahl and G. Stumme, Network properties of folksonomies. *AI Communications*. 20(4) (2007) 245-262.

[19] J. Sinclair and M. Cardew-Hall, The folksonomy tag cloud: when is it useful? *Journal of Information Science*. 34(1) (2008) 15-29.

[20] K. Lerman and L. Jones, Social browsing on flickr. *Proceedings of International Conference on Weblogs*

and Social Media, Boulder (Boulder, Colorado, USA, March 26-28, 2007). *ICWSM'07*.

[21] R. Van Zwol, Flickr: Who is Looking? *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence IEEE Computer Society* (Silicon Valley, California, USA, November 2-5, 2007). *WI'07* 184-190.

[22] R. Negoescu and D. Gatica-Perez, Topickr: flickr groups and users reloaded. *Proceedings of the 16th ACM International Conference on Multimedia* (Vancouver, British Columbia, Canada, October 26-31, 2008). *MM'08* (ACM Press, New York) 857-860.

[23] R. Negoescu, B. Adams, D. Phung, S. Venkatesh and D. Gatica-Perez, Flickr hypergroups. *Proceedings of the 17th ACM International Conference on Multimedia* (Beijing, China, October 19-24, 2009). MM'09 (ACM Press, New York) 813-816.

[24] H. Chen, M. Chang, P. Chang, M. Tien, W. Hsu and J. Wu, SheepDog: group and tag recommendation for flickr photos by automatic search-based learning. *Proceedings of the 16th ACM International Conference on Multimedia* (Vancouver, British Columbia, Canada, October 26-31, 2008). *MM'08* (ACM Press, New York) 737-740.

[25] J. Yu, X. Jin, J. Han and J. Luo, Social group suggestion from user image collections. *Proceedings of the 19th international conference on World wide web* (Raleigh, North Carolina, USA, April 26-30, 2010). *WWW 2010* (ACM Press, New York) 1215-1216.

[26] M. De Choudhury, H. Sundaram, Y. Lin, A. John and D. Seligmann, Connecting content to community in social media via image content, user tags and user communication. *Proceedings of the 2009 IEEE international conference on Multimedia and Expo* (New York, NY, USA, June 28-July 3, 2009). *ICME'09* (IEEE Press, Piscataway, NJ) 1238-1241.

[27] *Orkut*. Available at: http://www.orkut.com/ (accessed 14 December 2009).

[28] E. Spertus, M. Sahami and O. Buyukkokten, Evaluating similarity measures: a large-scale study in the orkut social network. *Proceedings of the 11th ACM SIGKDD international conference on Knowledge discovery in data mining* (Chicago, Illinois, USA, August 21-24, 2005). *KDD'05* (ACM Press, New York) 678-684.

[29] W. Chen, D. Zhang and E. Chang, Combinational collaborative filtering for personalized community recommendation. *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining* (Las Vegas, Nevada, USA, August 24-27, 2008). *KDD'08* (ACM Press, New York) 115-123.

[30] W. Chen, J. Chu, J. Luan, H. Bai, Y. Wang and E. Chang, Collaborative filtering for orkut communities: discovery of user latent behavior. *Proceedings of the 18th international conference on World wide web* (Madrid, Spain, Aprial 20-24, 2009). *WWW 2009* (ACM Press, New York) 681-690.

[31] M. Deshpande and G. Karypis, Item-based top-n recommendation algorithms. *ACM Transactions on Information Systems (TOIS)*. 22(1) (2004) 143-177.

[32] B. Sarwar, G. Karypis, J. Konstan and J. Riedl, Application of dimensionality reduction in recommender system–a case study. *ACM WebKDD 2000 Web Mining for E-Commerce Workshop*.

[33] S. Zhang, W. Wang, J. Ford and F. Makedon, Learning from incomplete ratings using non-negative matrix factorization. *Proceedings of the 6th SIAM Conference on Data Mining* (Bethesda, Maryland, April 20-22, 2006). *SDM'06* (SIAM Press) 548–552.

[34] P. Symeonidis, A. Nanopoulos and Y. Manolopoulos, A unified framework for providing recommendations in social tagging systems based on ternary semantic analysis. *IEEE transactions on knowledge and data engineering*. 22(2) (2010) 179-192.

[35] G. Adomavicius and A. Tuzhilin, Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE transactions on knowledge and data engineering*. 17(6) (2005) 734-749.

[36] D. Lee and H. Seung, Algorithms for non-negative matrix factorization. *Advances in neural information processing systems* (2001) 556-562.

[37] J. Carroll and J. Chang, Analysis of individual differences in multidimensional scaling via an N-way generalization of “Eckart-Young” decomposition. *Psychometrika*. 35(3) (1970) 283-319.

[38] D. Lee and H. Seung, Learning the parts of objects by non-negative matrix factorization. *Nature*. 401(6755) (1999) 788-791.

[39] A. Smilde, R. Bro and P. Geladi, *Multi-way analysis with applications in the chemical sciences* (Wiley, West Sussex, England, 2004).

[40] *Flickr API*. Available at: www.flickr.com/services/api/ (accessed 14 December 2009).

[41] Y. Koren, Factorization meets the neighborhood: a multifaceted collaborative filtering model. *Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining* (Las Vegas, Nevada, USA, August 24-27, 2008). *KDD’08* (ACM Press, New York) 426-434.

[42] T. Kolda and J. Sun, Scalable tensor decompositions for multi-aspect data mining. *In IEEE International Conference on Data Mining* (Pisa, Italy. December 15-19, 2008). *ICDM’08*.

[43] Z. Huang, D. Zeng and H. Chen, A comparative study of recommendation algorithms in e-commerce applications. *IEEE Intelligent Systems*, 22(5) (2007) 68-78.