Tag recommendation by machine learning with textual and social features

Xian Chen · Hyoseop Shin

Received: 5 February 2011 / Revised: 4 January 2012 / Accepted: 5 March 2012 /
Published online: 1 April 2012

Abstract Tags are very popular in social media (like Youtube, Flickr) and provide valuable and crucial information for social media. But at the same time, there exist a great number of noisy tags, which lead to many studies on tag suggestion and recommendation for items including websites, photos, books, movies, and so on. The textual features of tags, likes tag frequency, have mostly been used in extracting tags that are related to items. In this paper, we address the problem of tag recommendation for social media users. This issue is as important as the tag recommendation for items, because the tags representing users are strongly related to the users' favorite topics. We propose several novel features of tags for machine learning that we call social features as well as textual features. The experimental results of Flickr show that our proposed scheme achieves viable performance on tag recommendation for users.

Keywords Tag recommendation · Textual features · Social features · Machine learning · Social media · Flickr

1 Introduction

Recently, tags have become more and more important in social media, such as delicious, Flickr, Zooomr, and Youtube. In these communities, Users can freely use some keywords to annotate web pages, photos, and videos. This kind of keywords is called tags. For example, in Flickr, users can upload personal photos and use some tags to annotate their photos. For the motivation (Ames and Naaman 2007) why users use tags, we can see that people want to make their photos easily retrievable,

X. Chen · H. Shin (✉)
Web Intelligence Laboratory, Konkuk University, Seoul 143-701, South Korea
e-mail: hyoseop.shin@gmail.com

X. Chen
e-mail: chenxian@konkuk.ac.kr

or allow other users with similar hobbies to easily find them; at the same time, they want to easily search the photos in which they are interested through queries, or easily find the users who have similar hobbies. However, when the users tag items, not all tags used are related to the items. Users also use some non-meaningful tags to annotate the items: sometimes they use year, month, and people names, while other times they use the tags they created, like(“HelloMyLove”), Sometimes they use their own denotations of which other people cannot understand the meanings except themselves, (like”####”.) (Suchanek et al. 2008; Bischoff et al. 2008), and often they make spelling mistakes. All of the situations described above lead to the reality that most tags are irrelevant and noisy. Therefore, nowadays many studies on finding key phrases, key words, high quality tags (Sen et al. 2009a), tag suggestions, and tag recommendations related to items are very popular. A tag is defined as high quality if it helps the community understand an important aspect of an item (Sen et al. 2009a). Several papers (Sen et al. 2009a; Yih et al. 2006; Liu et al. 2009; Sigurbjörnsson and Zwol 2008; Witten et al. 1999) provide methods for finding high quality tags, key words or key phrases in websites, movies, photos, and documents. Most of them only used the textual features of tags, which represented the traditional computational linguistic research, such as the number of tags which are used by each user, TF and IDF. However, in our paper, we not only used textual features, but also used social features which related to the users’ social activities. In terms of users’ social activities, such as marking items as favorite, rating items and so on, we extract several novel features of tags that we call social features of the tags.” We base our research on Flickr users and their social activities. Using both social features and textual features, we propose to extract the *representative tags* of users which are defined as users’ tags that can represent for the users’ interested topics or can be related to users’ favorite items. In Flickr, users can not only upload photos, but they can also mark other users’ photos as their own favorite. Marking favorite photos is one kind of social activities in Flickr. From this social activity, we can conclude the users’ favorite topics. In our previous work (Chen and Shin 2010), we briefly described our method. First, we find the effective textual features and social features of tags. Second, we do machine learning to extract the representative tags of users. Furthermore, through the extracted representative tags, we can recommend interest groups, items, and other users to social community users who have similar hobbies and items. Here, in this paper, we describe the proposed method in more detail, and in particular, we present empirical results and discussions in depth on several important aspects such as the effect of each feature, social features vs. textual features, and frequency-based feature vs. ratio-based feature.

The rest of this paper is organized as follows. In Section 2, we describe related works. In Section 3, we propose the features of tags that we use in our research. In Sections 4, we describe the algorithm and training model. In Section 5, we provide the experimental environment and our user study system. In Section 6, we present our experimental results and analysis. Finally, we summarize the conclusion and our future work in Section 7.

2 Related works

There have been a large number of studies on tag suggestions and tag recommendations. Most studies focused on the relationship between tags and items, such as tags and contents, tags and websites, tags and photos, and tags and movies (Suchanek et al. 2008; Bischoff et al. 2008; Sen et al. 2009a; Yih et al. 2006; Liu et al. 2009; Sigurbjörnsson and Zwol 2008; Witten et al. 1999; Wu et al. 2009; Garg and Weber 2008; Lu et al. 2009; Heymann et al. 2008; Song et al. 2008). Suchanek et al. (2008) provided metrics to analyze the meanings of each tag, and they then proposed several different tag suggestion methods for contents of web pages; Bischoff et al. (2008) described whether or not all tags could be used for searching different items; (Yih et al. 2006; Witten et al. 1999) used textual features to find keywords for websites and documents; Sen et al. (2009a) studied the method to find high-value tags through tags' textual features, such as tag frequency. Liu et al. (2009) proposed a tag ranking scheme to rank the tags that are associated with a given photo according to their relevance to the photo content; Sigurbjörnsson and Zwol (2008) used some algorithms like Jaccard similarity to calculate the co-occurrence of tags for the purpose of suggesting related tags to the photos; Wu et al. (2009) provided a tag rank with a visual features approach for recommending tags based on photos; Garg and Weber (2008) proposed a tag co-occurrence algorithm to rank the tags of each photo; Lu et al. (2009) described an approach to search similar web pages through the co-occurrence of their tags; Heymann et al. (2008) discussed whether social tags can be applied to a particular object. Song et al. (2008) suggested a real-time automatic tag recommendation to documents.

As described above, we can see the previous work mainly focus on the relationship between tags and items. Meanwhile, it should be noted that, in social media, different users can have different characteristics, different hobbies, and different tastes, and thus tags can also be related to users as well as to items. There are several studies on the relationship among users, tags, and items. Wu et al. (2006) proposed schemes to support personalized web page search by considering the relationships among users, tags, and web pages in a probabilistic framework. Zhang et al. (2011) presented an approach that employing a feature correlation graph to capture the correlations between different features in an integrated manner and then evaluating tags' relevance to query image. Stoyanovich et al. (2008) proposed schemes to generate the URL hotlist (i.e., most interesting URLs) for each user. In del.ico.us, tags are used to annotate URLs by each personal user and his/her friends. They generated a series of formula to generate hotlists based on the overlaps in tags and URLs between a user and his/her friends. For example, the URLs that are tagged with the same tag more than a threshold by one's friends are selected and then scored by the number of users who tagged on that URL. Finally, top-scored URLs are included into the user's hotlist. The limitation of these methods is that some proposed metrics such as the tag frequency of each URL and tag co-occurrence between a user and his/her friends are evaluated against arbitrary threshold values, which can exclude many useful tags or items. In contrast, we present a machine learning-based approach in tag recommendation to efficiently combine textual features as well as social features.

According to the previous work, there has been limited research on tag recommendation for users. Giannakidou et al. (2011) presented a method for tracking macroscopic and microscopic users' interests, detecting emerging trends and recognizing events through tag clusters. Sen et al. (2009b) explored methods for finding users' preference tags for a movie website (i.e., MovieLens). At first, they used textual features (e.g., tag frequency) to find high quality tags for each movie. Then, they inferred the preference tags for each user depending upon the movies that are tagged or rated by this user. Meanwhile, in our paper, we are concerned with the users' *representative tags* that can represent users' favorite topics or topics of interest. In comparison, even though our proposed textual features could produce high-quality tags described in Sen et al. (2009b), the representative tags in this paper may not be comparable to the users' preference tags of Sen et al.'s work; in our paper, the content items themselves (e.g., photos) are assumed to be created by the users, but according to Sen et al. (2009b), only the reviews and ratings of the content items (e.g., movies) are generated by the users. Thus, the features of the users' representative tags should be different from the features of users' preference tags.

3 Features extraction

For finding out the representative tags for users that are strongly related to the users' interests, first, we extract the tag features that can effectively and significantly eliminate the noisy tags. We do not only use the textual features, but also the social features of tags. Based on linguistic analysis, we propose textual features that are directly extracted from each user's own terms in tags, titles, contents, and comments which are applied to his/her own photos. If users use some tags several times in titles, contents or comments, we can infer that users are interested in the topics on these tags. Also, if users use some tags frequently, we can infer that these tags may be related to their favorite topics. Social features are defined as the features that are extracted from each user's social activities. For instance, a user marks other users' photos as his/her favorites, or another user marks his/her own photos as favorites. If users marked other several photos which are related to the same tags, we can also infer that users are interested in the topics on these tags. Through both linguistic analysis and social activities, we collect features for tags of each user. Here, we describe the features proposed in this paper.

3.1 Textual features

- *tf* (tag frequency): The frequency of tag t_i for user u. the formula of *tf* is:

$$tf = n_i \tag{1}$$

  While n_i is the number of tag t_i used by each user.

- *tf_ratio* (tag frequency ratio): The ratio of the frequency for tag t_i against the sum of the frequencies of all the tags for each user, defined as follows:

$$tf\_ration = \frac{n_i}{\sum_k n_{k,u}} \tag{2}$$

  Where n_i is the number of occurrences of tag t_i for each user, and the denominator is the sum of occurrences of all the tags for one user. K means there are k different tags used by one user.

Features *tf* and *tf_ratio*, these features represent how often each user used the tag t_i among all the tags which he/she used. The more frequently the user used tag t_i, the more possible the tag t_i is related to the user's favorite topic.

- *iuf* (inverse user frequency): it is obtained by dividing the total number of users in the database by the number of users who used tag t_i, and then taking the logarithm of that quotient, defined as follows:

$$iuf = \log \frac{|U|}{|\{u : t_i \in u\}|} \tag{3}$$

  While $|U|$ is the total number of all users in our database, $|\{u : t_i \in u\}|$ is the number of users who used tag t_i.

However, users sometimes also use irrelevant tags, no-meaningful tags or tags that they made up themselves, but other people cannot understand most of the time. In this case, feature *tf* or *tf_ratio* values are also high. If we only depend on *tf* or *tf_ratio*, then the result will be deviated by this kind of non-meaningful tags. In order to prevent this deviation, we use *iuf* that indicates the popularity of tag t_i. The number of users who used the tag t_i can distinguish whether the tag is normal or rare. We can reduce strange and rare tags according to feature *iuf*.

- *in_title*: the number of tag t_i that appears in the titles (Yih et al. 2006) of the posts for a user. People usually use brief keywords in the titles of posts. It is defined as:

$$in\_title = \sum_k n_{k,ti} \tag{4}$$

  While $n_{k,ti}$ is the number of tags t_i that appeared in $k-$th title for each user.

- *in_content*: the number of tag t_i that appears in the contents of the posts of a user. When a user uploads his photos, sometimes he would give some descriptions about the photos that we call content. It is defined as follows:

$$in\_content = \sum_k n_{k,ti} \tag{5}$$

  While $n_{k,ti}$ is the number of tag t_i that appears in $k-$th content for each user.

- *in_comment*: the number of tags t_i that appear in the comments of the posts for a user (Shin et al. 2008). We can see that the users usually establish friendships with others who have similar favorite topics. Therefore, they may exchange their opinions about each other photos in the comments. It is defined as:

$$in\_comment = \sum_k n_{k,ti} \tag{6}$$

  While $n_{k,ti}$ is the number of tag t_i that appears in $k-$th comment for each user.

For features *in_title* and *in_content*, users always want to emphasize or express something essential about their photos through titles and contents. So, if the tag t_i appears several times in titles or contents, then it can be at least high quality tag for each user. For feature *in_comment*, users who uploaded the photos always discuss the photos with friends, or the user's friends want to express something related to the photos through comments. So, the feature *in_comment* may describe the relationship between the authors and the commentators. Therefore, we can infer that if the tags are important for the photos, then they may appear in the comments several times.

3.2 Social features

- *fav_coocc_freq* (favorite co-occurrence frequency): In Flickr, the users can maintain the list of *favorite* photos that are uploaded by other users. It can be inferred that two users will have common interests if they have common favorite photos. *fav_coocc_freq* represents the number of favorite photos of user u that contain the tag t_i. It is defined as:

$$fav\_coocc\_freq = p_{f,ti} \tag{7}$$

  While $p_{f,ti}$ is the number of photos that are marked by one user as favorite and also contain tag t_i.

- *fav_coocc_ratio* (favorite co-occurrence ratio): an alternative representation of *fav_coocc_freq* for considering the ratio against all the favorite photos, which is defined as follows,

$$fav\_coocc\_ratio = \frac{p_{f,ti}}{p_f} \tag{8}$$

  While p_f is the number of favorite photos of one user, $p_{f,ti}$ is the number of favorite photos of this user that contain tag t_i.

The way to use social features has an impact on the methods used to extract knowledge for the information of users' social activities. From Fig. 1, we can notice that there are some relationship between users and their favorite topics which are represented by their social activities and tags. And also we can see the topics of users' favorite photos overlap with topics of their own photos, because when they have interested in some topics, they not only upload their own photos about the topics, but also pay attention to the photos which are uploaded by other users but related to their interested topics. Therefore by using social features, we can get significant information for personal users. In Flickr, users can not only upload their own photos and mark others' photos as their own favorite, but also they can use tags to tag their own photos or others' photos. Figure 1 shows the process of extracting fav_coocc_freq and fav_coocc_ratio. The user u has his/her photos and tags that are used to describe his/her photos, and user u has several favorite photos. These features consider the number of co-occurrence of a given tag. Suppose that user u had five favorite photos and used several tags, ”nature”, “flowers”, “happy”, “beautiful” and “landscape”. Tag “nature”, “flowers”, “landscape” also appeared in user u's favorite photos.

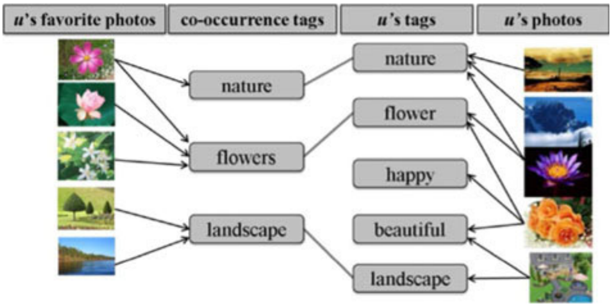


Fig. 1 fav_coocc_freq and fav_coocc_ratio example

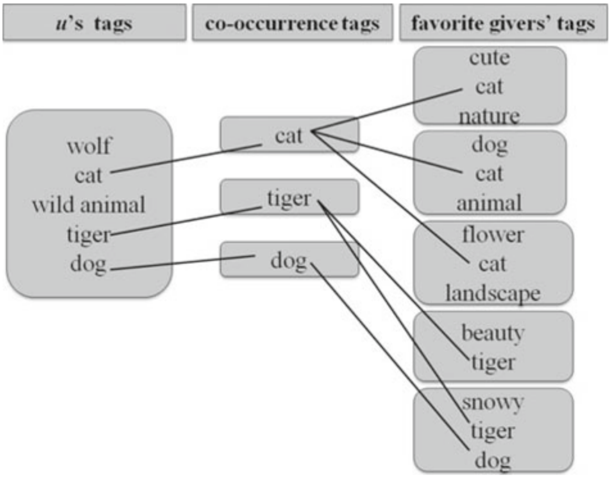


Fig. 2 fav_giver_freq and fav_giver_ratio example

Tag “nature” appeared in one of user u’s favorite photos, so the fav_coocc_freq and fav_coocc_ratio values for “nature” are one and 1/5; Tag “flower”appeared in three favorite photos of user u, so the fav_coocc_freq and fav_coocc_ratio values for “flower” are three and 3/5; Tag “landscape” appeared in two favorite photos, so the fav_coocc_freq and fav_coocc_ratio values for “landscape” are two and 2/5.

- *fav_giver_freq* (favorite giver frequency): the number of users who marked at least one of user u’s photos as favorite and also used the tag t_i in their own photos. It is defined as:

$$fav\_giver\_freq_{ti} = fg_{ti} \tag{9}$$

While fg_{ti} is the number of one user’s favorite givers[1] who also used the tag t_i.

- *fav_giver_ratio* (favorite giver ratio): an alternative representation of *fav_giver_freq* for considering the ratio. The ratio is obtained by counting only the users who used the tag t_i among all users who marked at least one of user u’s photos as favorite, which is defined as follows,

$$fav\_giver\_ratio_{ti} = \frac{fg_{ti}}{fg} \tag{10}$$

The denominator is the number of favorite givers of one user, and the numerator is the number of this user’s favorite givers who used tag t_i.

Figure 2 shows the process of how to get the values of *fav_giver_freq* and *fav_giver_ratio*. Suppose user u has several tags and five favorite givers with their own tags, “wolf”, “cat”, “wild animal”, “tiger”, and “dog”. For the co-occurrence tags between user u and his/her favorite givers, tag “cat” was also used by three favorite givers, and thus the *fav_giver_freq* and *fav_giver_ratio* values for “cat” are three and 3/5; Tag “tiger” was used by two favorite givers, so the *fav_giver_freq* and

[1] Favorite givers: We call the users who marked user u’s photos by favorite as favorite givers of user u.

fav_giver_ratio values for "tiger" are two and 2/5; in the same way, the *fav_giver_freq* and *fav_giver_ratio* values for tag "dog" are one and 1/5.

For social features, *fav_coocc_freq* and *fav_coocc_ratio*, these features directly point out each user's favorite photos. From his/her favorite photos, we could conclude his/her hobbies or the topics that he/she is interested in, and these photos are connected with several tags. Therefore, among all the tags in one user's favorite photos, the more frequently the tag t_i appeared, the more possible the tag t_i can represent the favorite topics of user u. For features *fav_giver_freq* and *fav_giver_ratio*, among each user's favorite givers, the more favorite givers used the tag t_i that one user also used, the more possible that the user and his/her favorite givers will have the same favorite topics that are related to the tag t_i.

4 Algorithm

To select representative tags for each user, we provide several features for candidate tags, and through a machine learning algorithm, we then build a model for the training set. During our experiments, we examined several different machine learning algorithms such as logistic regression, neural network and Naïve Bayes. Detail information is shown in Table 1. According to the experimental results, we chose the Naïve Bayes Classifier that achieved the best results among all the algorithms.

Hence, according to the experimental results, we chose the Naïve Bayes Classifier that achieved the best results among all the algorithms. When we used the machine learning algorithms with our features to find out the representative tags for user u, we marked whether or not the tag t_i is a representative tag. Then, the class is a binary feature for our scheme. For all equations, "Y" means the probability that the candidate tag can be representative of users under a situation of different tag features. Further, F means all the features that are used for our training and testing. The formulas are as follows:

$$P(Y|F_1, \dots F_n) = \frac{P(Y)\,P(F_1 \dots, F_n|Y)}{P(F_1, \dots, F_n)} \tag{11}$$

$$\begin{aligned} &P(Y)\,P(F_1, \dots, F_n|Y) \\ &\quad = P(Y)\,P(F_1|Y)\,P(F_2, \dots, F_n|Y, F_1) \\ &\quad = P(Y)\,P(F_1|Y)\,P(F_2|Y, F_1)\,P(F_3, \dots, F_n|Y, F_1, F_2) \\ &\quad = P(Y)\,P(F_1|Y)\,P(F_2|Y, F_1) \dots P(F_n|Y, F_1, F_2, \dots, F_{n-1}) \end{aligned} \tag{12}$$

Table 1 Performance comparison with other machine learning algorithms

Algorithms	Precision	Recall	F-score
MLP	57.9%	17.9%	27.4%
Logistic regression	50.8%	4.5%	8.3%
Naïve Bayes	49%	54.5%	51.6%

$$p(Y)P(F_1,\ldots,F_n|Y) = P(Y)P(F_1|Y)\,P(F_3|Y)\ldots P(Y)\prod_{i=1}^{n} p(F_i|Y) \tag{13}$$

$$P(Y|F_1,\ldots F_n) = P(Y)\prod_{i=1}^{n} p(F_i|Y) \tag{14}$$

$$P(Y) = \frac{y}{y+n} \tag{15}$$

Based on Bayes' theorem, we can get (11). In practice, we are only interested in the numerator of (11), since the denominator does not depend on Y and the values of the features F_i are given, and thus the denominator is effectively constant. Hence, we can get (12), and because each feature F_i is assumed to be conditionally independent of every other feature F_j for $j \neq i$, we get (13). Finally, we use (14) for our algorithm. In (15), y and n represent the number of positive instances and the number of negative instances, respectively, in our training data. Positive instances mean that the candidate tag t_i is representative of user u; on the contrary, negative instances means that the tag t_i is not a representative tag.

5 Experimental environments

5.1 Data set

We conducted our experiments using the data set of Flickr (2000). We collected 813, 353 users who registered on Flickr and downloaded their photos with tags for the whole month of Dec, 2009. However, among the latter Flickr users, there were 262,722 users who applied tags to their photos. Therefore, we built our dataset with these 262,722 users. Finally, we have 10,591,157 photos, 1,600,349 tags and 4,828,926 favorite feedbacks for all the users. Table 2 provides the basic information about our data set. In our database, the maximal number of tags assigned by one Flickr user is 3,702 and the average number of tags assigned by Flickr users is 52.

5.2 Data preprocessing

In the data set, we noticed that there are many tags that had the same meanings but the formats were different. For example, "tree", "trees", "Tree", and "Trees" would be identified as the same tag. Another example is "nature finest", "NatureFinest", and "Nature finest". Therefore, we preprocessed the same tags before carrying out

Table 2 Experiment data set from Flickr in Dec. 2009

Experiment data	Number
Total number of users	262,722
Total number of photos	10,591,157
Total number of tags	1,600,349
Total number of favorite feedbacks	4,828,926
Maximum number of tags per user	3,702
Average number of tags per user	52

our experiments. We added a blank between two words if there was no space; we removed the plural suffix formats and changed the upper cases to lower cases. In the previous examples, the first tags were grouped as “tree”, and the second ones were as “nature finest”.

We noticed that the tags that consist of more than four words (e.g., “a room with a view”, “a sad little Christmas tree”, “a picture of Jack a day”) are not often used by users. All of these tags seem to be described for users’ special situations, and most of them have been only applied by once. Therefore, in our experiments, we do not consider the tags that contain more than four words. Moreover, most of tags are composed by single words, and there are not too many tags as sentences. In data pre-processing step, we lemmatize tags and process stopwords. Table 3 shows the number of stopwords among the tag instances in our experiment; only a small portion of tag instances are stopwords. Actually, in the user study, none of single tags which are stopwords were chosen as a representative tag. Only exception was “the unforgettable picture”, which was chosen as representative tags.

5.3 User study

In our user study, since the data set was too large to evaluate the tags of all users that we collected, we chose 177 users among the entire database. In the whole database, we first randomly chose 250 candidate users that satisfied the following conditions: the number of posts was more than 10; for each post, the number of tags was more than 10 but less than 15; at the same time, the favorite number of each post was more than 10. However, there were some users whose photos we could not see (maybe they had already removed some). Then we removed these users. Finally, we had 177 candidate users with 1,770 photos and 13,464 tags for training and testing. We asked seven evaluators to choose the representative tags for each user. In our evaluation system, we only show the top 10 most popular photos and the tags that only have less than four words and only appeared in these 10 photos for each user. We showed the users’ photos and tags at the same time to the evaluators who were responsible for choosing the representative tags for each user. First, evaluators watch the photos for each user; after they watch the photos, they should have their own opinions for each user, such as what kind of photos does this user mostly uploaded, girls or landscape? What is the main favorite trend for this user?; Second they conclude the types of photos which can represent each user’s most favorite topics; Finally, they choose the tags which showed to them and can be representative for each user’s interested topics. Each tag was then marked as a representative tag or non-representative one for each user. This binary feature is the class feature used by the machine learning scheme, and its value has just “yes” or “no”. With the evaluation results, we used the Weka (2001) program to experiment on training, testing, and prediction.

During our user study, we asked these seven evaluators to evaluate all users. Therefore, for each tag of each user, these seven evaluators may have different opinions. In our experiments, if there were three or more than three evaluators who

Table 3 Tags with stopwords

Total tag instances	13464
Tag instances which included stop words	727
Tag instances which have one word but also stop words	101

Table 4 User study results

User study	Number
Total number of candidate users	177
Total number of photos of candidate users	1,770
Total number of tags of candidate users	13,464
Total number of evaluators	7
Total number of representative tags chosen by evaluators	1,355
Average number of representative tags per user	7.7

chose the tag as representative tag, then we marked this tag as a representative tag. The reason why we choose three as the threshold of representative tags is that: the evaluators only have two choices, if evaluators choose one tag, which means they consider this tag can be representative for users' interested topics. Otherwise, if evaluators do not choose one tag, which means they consider this tag cannot be representative tag for users. If we use agreement as one, or two, which means, six, or five evaluators think this tag cannot be the representative tag. This is the reason why we do not consider about agreement equal to one or two. However, agreement would be used as four, which is more than average number of evaluators, we consider about the evaluators sometimes may ignore or skip some tags which can be representative tags because of their carelessness. Therefore, we use agreement as three. Finally, there were 1,355 tags marked as representative tags among 13,464 tags. Table 4 shows the results of our user study. For 177 users, the average number of representative tags is 7.7, which is much smaller than the average number (i.e., 52) of tags generated by Flickr users. This implies that our method can be quite effective to extract representative tags for Flickr users.

In order to measure the agreement among the evaluators, we used Kappa statistic (Phan et al. 2010) to examine the agreement among the evaluators. We calculated the Kappa values between every two evaluators, and then did the average as follows:

$$K = \frac{1}{C_2^N} * \sum\nolimits_{x\in U} \sum\nolimits_{y\in U\neq} \frac{P_A(x,y) - P_E(x,y)}{1 - P_E(x,y)} \tag{16}$$

Where N is the number of evaluators, P_A(x, y) denotes of the times that two evaluators, x and y, agreed, $P_E(x, y)$ denotes the proportion of the times that two evaluators, x and y, would agree by chance. In our user study, the Kappa statistic was 60.3%, which is a not bad value.

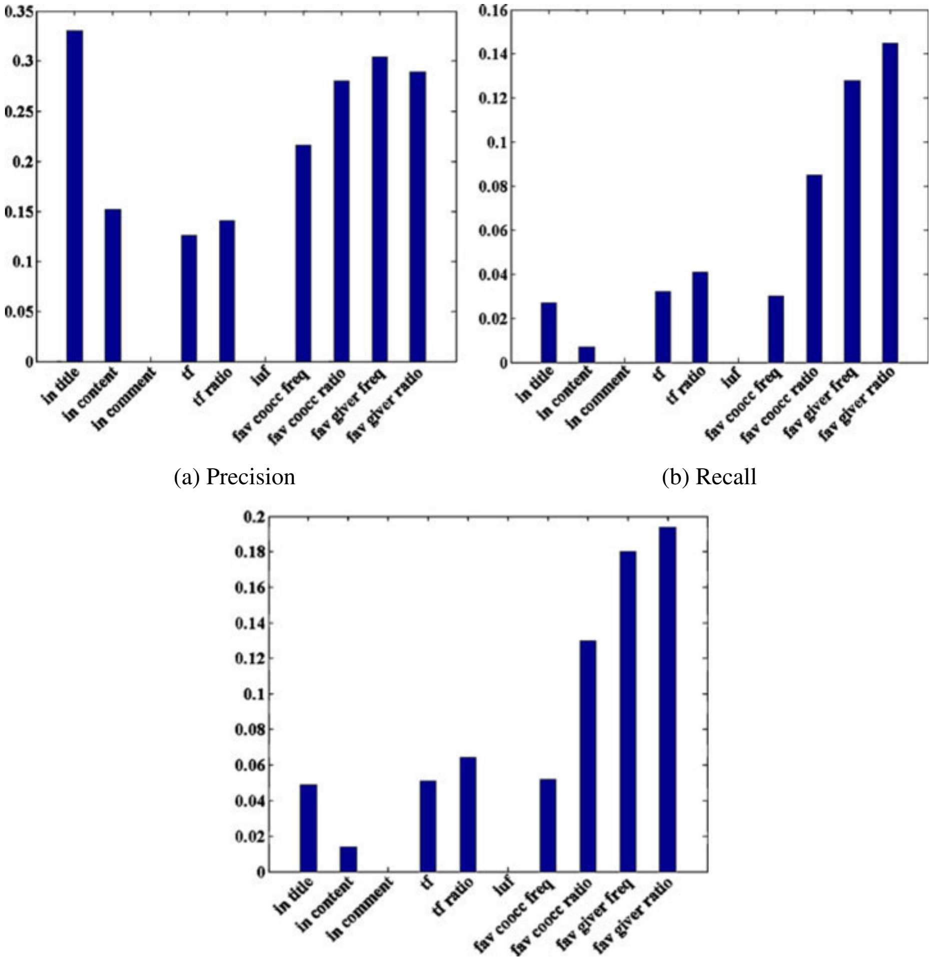
We discreted all the values of features and experimented with several machine learning algorithms. We separated the training and testing data in 10 fractions, and we alternately use nine of 10 for training and left one for testing. Finally, we take the average of accuracies.

6 Evaluation

6.1 Effect of each feature

After the evaluators selected the representative tags for each of the 177 users, we calculated all the features' values for each tag. As we introduced above, there are 10 kinds of features in our research. Which feature worked best in our experiment?

How important is a particular type of feature for extracting the representative tags for users? We studied these problems by experimenting with each feature and determining how much effect each caused. Figure 3 shows each feature's Precision, Recall and F-score value. In our paper, for calculating the F-score, we used the balanced F-score which is the harmonic mean of precision and recall (F-score 2002). The F-score showed that the social features outperformed the textual features. Because the social features extracted representative tags for each user, which are related to user's favorite topics, these tags are also high quality tags. But for the textual features, they can only extract high quality tags for users, which not all of them can be representative of the users' favorite topics. However, the high precision values of textual features also tell us that textual features can distinguish the noisy



(a) Precision

(b) Recall

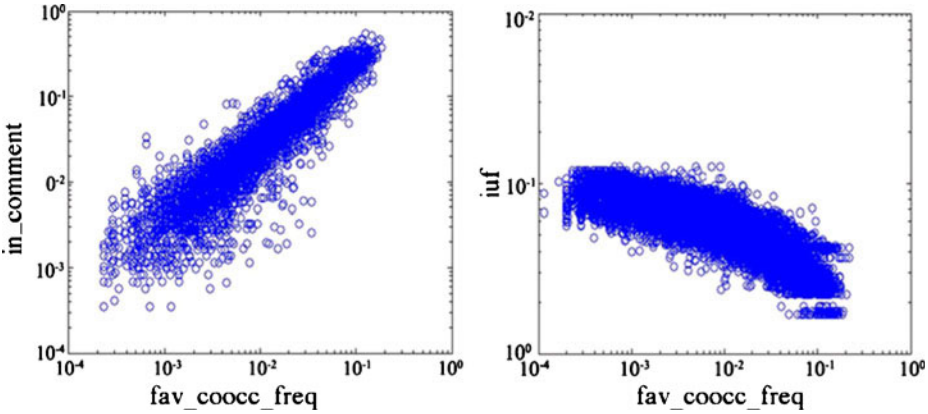
(c) F-score

Fig. 3 Precision, recall and F-score for each feature

tags efficiently. The low value of textual features also showed that most high quality tags cannot be representative for the users' favorite topics.

As for the social features, it is noted that *fav_giver_ratio* is much better than *fav_coocc_ratio*. The reason why *fav_giver_ratio* outperformed *fav_coocc_ratio* is that when a user watched some photos and thought they were nice or funny, user *u* marked these photos as favorite without considering whether or not they were related to his/her favorite topics. If these kinds of photos are too many, then the result will be deviated. On the contrary, for *fav_giver_ratio*, if most favorite givers of user *u* use the same tag used by user *u*, then we can infer that user *u* and his/her favorite givers have similar favorite topics related to the tag. As a result, *fav_giver_ratio* outperformed other features and received F-score of almost 20 %.

As for the textual features, we can see the *in_comment* and *iuf* were the worst. *iuf* represents the popularity of each tag. The results tell us that not all tags that are most popular or used by most users cannot be representative of different users' favorite topics. However, *iuf* can tell whether a tag is a common or rare tag. If there are many users who use this tag, then the *iuf* value will be low; on the contrary, if users seldom use this tag, then the *iuf* value will be high. Therefore, through *iuf*, we can distinguish a common tag from a rare tag. For *in_comment*, we found out why it worked worst. Figure 4a shows the distributions for both *in_comment* and *fav_coocc_ratio*. It is noted that there is a high correlation between these two features. We can find that when users comment on the photos, most of them will also mark them as their favorite. However, they may only write: "good", "nice photo", "I like it." As the same way, when users add their favorite, they also write some simple comments, such as "great!", "beautiful" and so on. Based on these comments, we cannot extract useful information from comments. It is also shown that traditional linguistic analysis also has its own drawback and limitation. However, when we transform linguistic analysis to social analysis, through different aspect, we can get unexpected better result. Hence, it is implied that *fav_coocc_ratio* can replace *in_comment*. As for *fav_coocc_ratio* and *iuf*, the greater the value of *fav_coocc_ratio* is, the smaller the value of *iuf* is. This indicates that the tags with high popularity



(a) *fav_coocc_ratio* and *in_comment* (b) *fav_coocc_ratio* and *iuf*

Fig. 4 The distributions of *in_comment* and *fav_coocc_freq*, *fac_coocc_freq* and *iuf*

are more likely to be representative of users than rare tags. Figure 4b shows the distribution of *fav_coocc_ratio* and *iuf*. It also tells that the tags which are used in most photos marked as users' favorite are also common and popular in the whole data set. That means several Flickr users must have common interested topics.

Among the features, we provided both frequency feature and ratio feature: *tf* and *tf_ratio*, *fav_coocc_freq* and *fav_coocc_ratio*, and *fav_giver_freq* and *fav_giver_ratio*. Figure 3 shows that the ratio features perform better than the frequency features when a single feature is considered. There is not much difference in performance between *tf* and *tf_ratio*, and between *fav_giver_freq* and *fav_giver_ratio*. However, the performance gap between *fav_coocc_freq* and *fav_coocc_ratio* is considerable. The reason can be inferred as follows; among the favorite photos of user *u*, there are some common objects appearing in some favorite photos, which are described by the same tags, but these tags, such as year, month or people's names, are not important. In this case, using *fav_coocc_ratio* can avoid these noisy tags that are applied for the common objects that are not important.

6.2 Social features vs. textual features

From Fig. 3, we can see that the social features outperformed textual features when only one feature is considered for extracting the representative tags. In this section, we will try to combine the textual features together, combine the social features together, and finally combine the textual features with social features to obtain the most effective feature set for our research.

Figure 5 shows the value of F-score for each feature set. For the textual feature sets, we have { *in_title*, *in_content*, and *in_comment*}, {*tf_ratio*, *iuf*}; for the social feature sets, we have {*fav_coocc_ratio*, *fav_giver_freq*}. For the combination of textual features and social features, we have {*tf_ratio*, *iuf*, *fav_coocc_freq*, *fav_giver_freq*, *in_title*, *in_content*, *in_comment*}, {*tf_ratio*, *iuf*, *fav_giver_freq*, *in_title*}. Among the textual feature sets, it is obvious that {*in_title*, *in_content*, and *in_comment*} is weak. Nevertheless, the F-score for the set of tf and iuf is almost 40%. For the social feature set, the F-score for {*fav_coocc_ratio*, *fav_giver_freq*} is about 31%. We not only tried the combination of *fav_coocc_ratio* with *fav_giver_freq*, but also another combination of social features, and we had similar results.

Sen et al. (2009a) has already showed that the textual features are effective at distinguishing high quality tags from noisy tags. However, the high quality tags

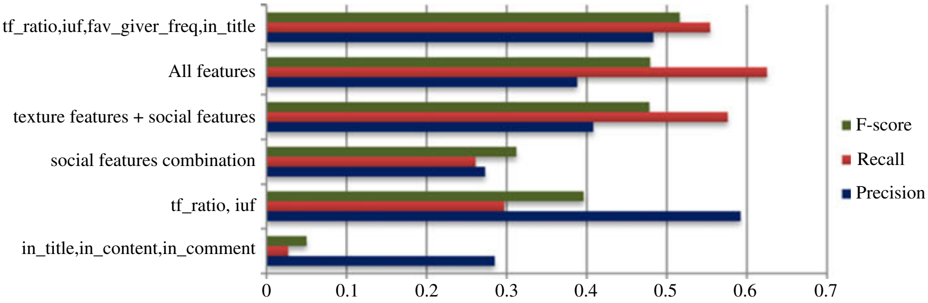


Fig. 5 F-score for feature combination

are different from the tags that can be representative of the users' favorite topics. Although the F-score of the set *tf_ratio* with *iuf* is higher than the F-score of the set of *fav_coocc_ratio* with *fav_giver_freq*, only the textual features are not likely to extract enough representative tags for each user that are related to their favorite topics. This is because the textual features mainly describe the linguistic characteristics of the tags; meanwhile, the social features describe the characteristics for the users' own favorites. As a consequence, we tried combinations of *tf*, *tf_ratio*, *iuf*, *fav_coocc_freq*, *fav_coocc_ratio*, *fav_giver_freq*, and *fav_giver_ratio*, and we chose the set of *tf_ratio*, *iuf*, *fav_coocc_freq* and *fav_giver_freq* from which we obtained precision of 0.408, recall of 0.576, and F-score of 0.478.

Finally, among the arbitrary feature combinations, we found that a combination of four features (*tf_ratio*, *iuf*, *fav_giver_freq*, and *in_title*) outperformed any single feature and feature sets, achieving precision of 0.49, recall of 0.545, and F-score of 0.516, which are shown in Fig. 6.

6.3 Effect of in_title, in_content and in_comment

Even though the result is not good when only *in_title*, *in_content*, and *in_comment* are used, we further test whether this feature set can make any sense when used with other features. We examine the experimental results for the feature set with *in_title*, *in_content* and *in_comment* and the feature set without them. We combine the feature set *tf_ratio*, *iuf*, *fav_coocc_freq* and *fav_giver_freq* with and without *in_title*, *in_content* and *in_comment*. According to the results shown in Fig. 7, with *in_title*, *in_content*, and *in_comment*, the recall is higher but the precision gets lower than without them. As a result, the F-score is almost the same. This implies that the features *in_title*, *in_content*, *in_comment* are not influential in extracting the representative tags for users.

6.4 Frequency feature vs. ratio feature

In this section, we compare the results between the social frequency features and social ratio features. In our research, we provided both frequency and ratio for several

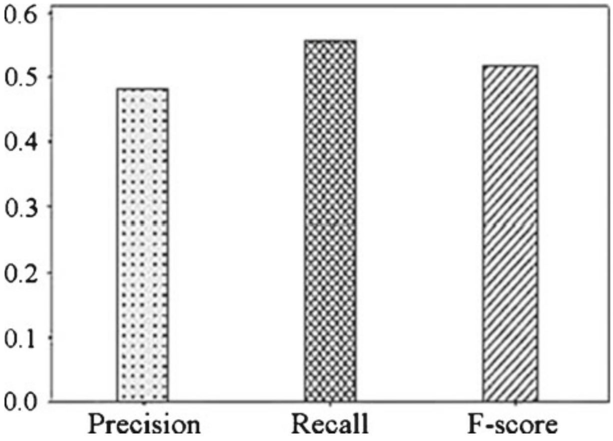


Fig. 6 Precision, recall, and F-score of the optimal feature set including *tf_ratio*, *iuf*, *fav_giver_freq* and *in_title*

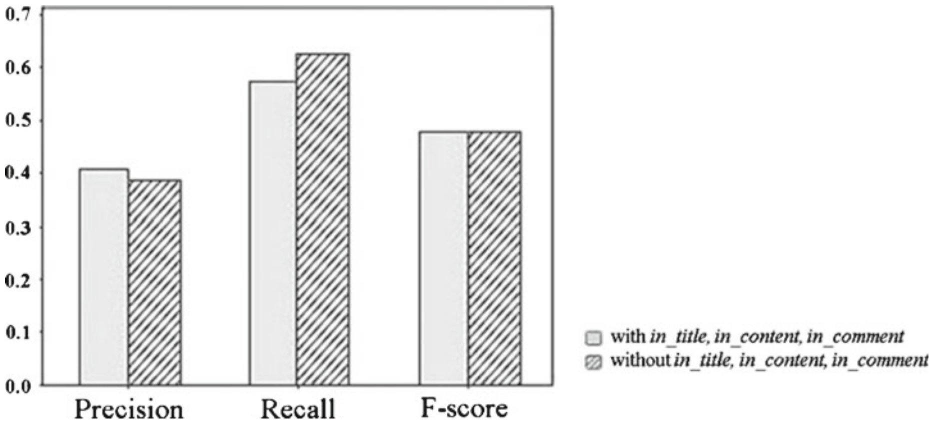


Fig. 7 Precision, recall, and F-score Performance between with and without *in_title*, *in_content*, *in_comment*. The *left solid bar* of each pair is the result without *in_title*, *in_content*, *in_comment*, while the *right dashed bar* is the result with *in_title*, *in_content*, *in_comment*

features, such as *tf* and *tf_ratio*, *fav_coocc_freq* and *fav_coocc_ratio*, *fav_giver_freq* and *fav_giver_ratio*. We notice that, when we use a single feature for an experiment, the ratio features outperformed the frequency features whether the feature uses is textual or social. However, when we combine the social features with textual features, the result is better when the social frequency features are used than that of social ratio features. Figure 8 shows the precision, recall, and F-score of the textual features with social frequency features and textual features with social ratio features. The precision of textual features with social ratio features is lower, but the recall for textual features with social ratio features is higher. As a result, the F-score of the textual features with social ratio features is lower than the textual features with social frequency features. The reason why it is better to use social frequency features is that

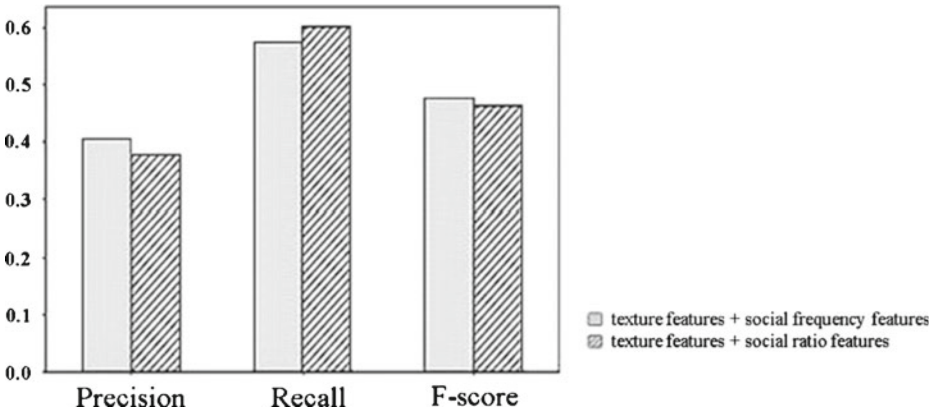


Fig. 8 Precision, recall, and F-score; the *left solid bar* of each pair is the result for textual features with social frequency features, and the *right dashed bar* is the result for textual features with social ratio features

the textual features are good at extracting high quality tags for items; when we use the textual and social features together, we can get high quality tags because of the textual features. Based on these high quality tags, it is enough to use the frequency social features; otherwise, if we use ratio social features, then it is too strict for the high quality tags that lead to reduction in accuracy. On the contrary, when only social features are used, strict ratio features are needed to identify the high quality tags.

6.5 Comparison between social features

In this section, we compare our two social features. Figure 9 shows the F-score for both *fav_coocc_ratio* and *fav_giver_ratio*. We have already explained why *fav_giver_ratio* worked better than *fav_coocc_ratio* in Section 6.1. Sometimes users just mark other users' photos as their favorite only because they think those photos are nice, pretty, or funny, without thinking whether or not these photos are related to their favorite topics. Under this situation, using *fav_coocc_ratio* is better than using *fav_giver_ratio*. Furthermore, in our optimal feature combination that consists of *tf_ratio*, *iuf*, *fav_giver_freq* and *in_title*, *fav_coocc_freq* and *fav_coocc_ratio* are excluded. It is implied that the social features *fav_giver_freq* and *fav_giver_ratio* describing the favorite givers of each user can play significant roles in the social network.

For the social features of tags, we not only consider the users' direct favorite items, but we also consider the relatioanship between the users and their favorite givers. And the most important discovery is that the features describing the relationship between users and their favorite givers are more effective than the features describing the users' direct favorite items. Consider the example of social network in Fig. 10. In social networks, users will typically have two types of relationships against the items. First, a user generates his/her own items that may receive favorites from other users (i.e., favorite givers). This is denoted by a solid red frame on the left. Second, a user may give favorites to items that are generated by other users(i.e., favorite receivers). This is denoted by the dotted purple frame on the right. Even though the favorite givers have been ignored in previous works, our experimental results show that our proposed *favorite giver* feature is more effective than the *favorite receiver* feature.

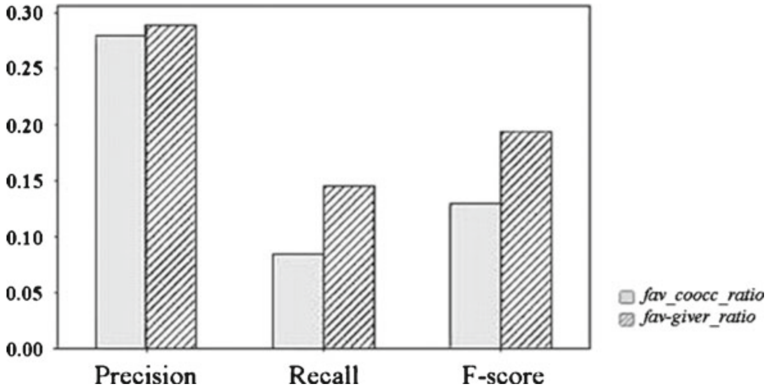


Fig. 9 Precision, recall and F-score for feature *fav_coocc_ratio* and *fav_giver_ratio*

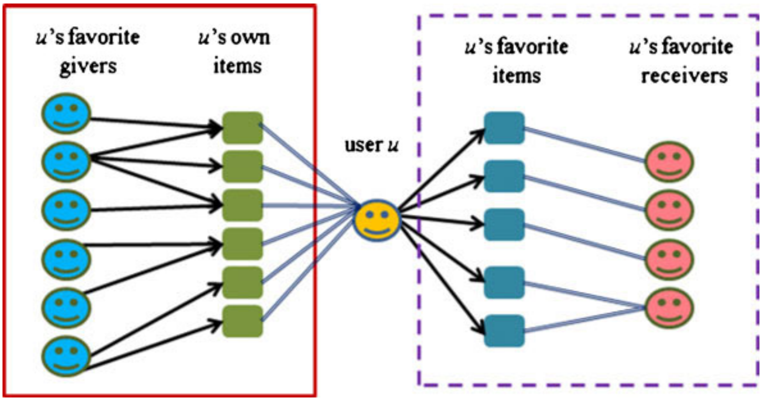


Fig. 10 Two types of relationships between users and items

6.6 Comparison with other methods

There are several research works with which we compare our results here. KEA (Witten et al. 1999) is an algorithm for automatic extraction of keyphrases from text. First, KEA processes documents by drewsting stopwords and stemming, and identifies candidate keyphrases. It proposed three feature values for each candidate, tf, idf, and first occurrence (number of words before the first occurrence of the phrase). The classifier is then trained by applying the Naïve Bayes algorithm. However, in photo sharing communities such as Flickr, there are not too many sentences and paragraphs, thus first occurrence is not available in our case. When we applied *tf* and *iuf* to our environment with Naïve Bayes, we achieved precision 59.2%, recall 29.7% and F-score 39.6%. We also tried first occurrence value when tags appeared in contents and obtained precision 61.3%, recall 28.9% and F-score 39.3%, which showed in Table 5.

Yih et al. (2006) extracted keywords for webpages. They proposed many linguistic features and use logistic regression for machine learning. The features are as follows:

(1) Lin: whether the candidate phrase is noun, noun phrase or not.
(2) C: whether the candidate phrase has capitalization or not.
(3) H: whether the candidate phrase appeared in hypertext or no.
(4) Ms: Whether the candidate phrase appeared in meta tags.
(5) T: whether the candidate phrase appeared in the HTML header.
(6) M: whether the candidate phrase is in the meta feature.
(7) URL: whether the candidate phrase is in the URL.

Table 5 Comparison with previous works

Methods	Precision	Recall	F-score
KEA (Witten et al. 1999)	61.3%	28.9%	39.3%
Find advertising keywords (Yih et al. 2006)	58.4%	11.1%	18.6%
Learning to recognize valuable tags (Sen et al. 2009a)	60.2%	22%	32.2%
Proposed method	48.3%	55.4%	51.6%

(8) IR: use TF and IDF.
(9) Len: the length in the sentence where the candidate occurs.
(10) PhLen: length of the candidate phrase.
(11) Query log: whether in the query or not.

Lin, H, T, IR and PhLen can be applied to our environment as they are. However, the other features are modified to adjust to our environment. For C, tags are consistently converted to lower case letters. For Ms and M, meta tags are meaningless in Flickr posts. For URL, there are no URL tags in our data set. Len is also not necessary because there are not many sentences in Flickr. For Query log, we cannot get this information from Flickr. Instead of T and H, we used *in_content* and *in_title*. Using logistic regression, we had precision 58.4%, recall 11.1% and F-score 18.6% (Table 5).

Sen et al. (2009b) infers users' preference tags in movie community. They recommend users' preference tags by using high quality tags and their own six algorithms. They did their experiments in the MovieLens website that is a movie recommendation service. First, they used the high quality tags that were extracted from (Sen et al. 2009a). In this step, they used several textual features of tags. At the second step, they used the fact that movies are rated by users and suggested six algorithms to infer users' preference tags. In comparison, this step cannot be comparable to our methods, for the content items themselves (e.g., photos) in Flickr are assumed to be created by the users, but only reviews and ratings for the content items (e.g., movies) in Sen et al. (2009b) are generated by the users. Therefore, we apply the features for finding high quality tags for comparison as follows:

(1) Num-item-apps: tags applied to a particular item by more than users are more relevant.
(2) Num-app: tags applied more time overall across items are more relevant.
(3) Num-users: tags applied overall across items by more users are more relevant.
(4) Num-searches: tags searched for more times are more relevant.
(5) Num-search-users: tags search for by more users are more relevant.
(6) Tag-share: tags that account for a larger fraction of an item's tag applications are more relevant.
(7) Avg-fraction-items-tagged: tags whose creators apply the tag many times are more likely to be list-making tags, and less relevant for the community as a whole.
(8) Apps-per-item: tags applied more often to the items to which they are applied are more relevant.
(9) Num-tag-words: tags with many words are less desirable.
(10) Tag-length: tags with very few letters are less desirable.

Among these, Num-item-apps, Tag-share, Num-searches and Num-search-users are not available in Flickr. Therefore, we applied the remaining six features with SVM to our environment and obtained the result: precision 60.2%, recall 22% and F-score 32.2%. Table 5 summarizes the performance comparison with the previous works.

6.7 Examining agreement levels

Table 6 summarizes the performance of the proposed scheme with respect to various levels of evaluators' agreement. We do not consider one or two agreement level

Table 6 Results of various agreement levels

Agreement	TP	TN	FP	FN	Precision	Recall	F-sscore
3	887	10629	1480	468	48.3%	55.4%	51.6%
4	464	11639	1075	286	30.1%	61.9%	40.5%
5	206	12200	894	164	18.7%	55.7%	28%
6	66	12803	495	100	11.8%	39.8%	18.2%
7	15	13197	228	24	6.2%	38.5%	10.6%

because they are too low. As shown in the table, as the agreement level gets higher, the number of TP (true positive) tags gets rapidly smaller because the agreement condition becomes stricter. Even though the number of TN (true negative) increases with higher agreement level, adopting too high a agreement level can cause to select too little number of representative tags, Our problem of extracting a relatively small number of representative tags among a large-scale tag pool is imbalanced in terms of the number of representative tags and the number of non-representative tags. This is the main reason that the performance gets worse when we increase the level of agreement too high. As a result, we conclude that medium-levels of agreement (i.e., three or four), is acceptable for a reasonable performance.

6.8 Validating with top 20% users

Our user study in Section 5.3 for training the algorithms applied a strict condition in selecting Flickr users, posts, and candidate tags in order to achieve better performance: we selected candidates among the users who wrote at least 10 posts, each of which contains 10 to 15 tags and is related to at least 10 users by a favorite link, and finally we chose 177 users for user study with 1,770 photos and 13,464 tags for training and testing. In this section, we validate the trained algorithm with different user data set. We rank Flickr users by the schemes (Shin et al. 2010) that can rank users in social networks with respect to their reputation and sociability. Then, we randomly choose 86 users among the top 20% users. We suppose that top 20% users can cover a considerable part of meaningful users who write posts and tags in Flickr. Table 7 summarizes our experiments with top 20% users. As a result, we obtain 43.1% precision, 52.9% recall and 47.5% F1-score. The results are not quite different from the results on our original data set, and thus it is conjectured that our proposed scheme can be applied to arbitrary meaningful users.

Table 7 Experiments with top 20% users

User study	Number
Total number of candidate users	86
Total number of photos of candidate users	8,634
Total number of tags of candidate users	3,152
Total number of evaluators	7
Total number of representative tags chosen by evaluators	769
Average number of representative tags per user	8.9

7 Conclusion

In this paper, we presented a scheme to effectively find representative tags for users that are related to the users' favorite topics in large-scale online communities. We use both textual features and social features of tags, and we use the Naïve Bayes Classifier algorithm to find the representative tags that can be related to the users' favorite topics. Our experiments showed that the textual features are good at finding high quality tags for items, but the social features can help us to better extract tags that are directly relevant to the users' topics of interest. Our research can be used, in finding users' groups who have the same favorite topics or in recommending potential interest groups to users, based on users' representative tags.

Acknowledgement This paper was supported by Konkuk University in 2010.

References

Ames, M., & Naaman, M. (2007). Why we tag: Motivations for annotation in mobile and online media. In *Proceedings of the SIGCHI conference on human factors in computing system* (pp. 971–980). California: ACM Press.

Bischoff, K., Firan, C. S., Nejdl, W., & Paiu, R. (2008). Can all tags be used for search? In *Proceedings of 17th ACM conference on information and knowledge management* (pp. 193–202). California: ACM Press.

Chen, X., & Shin, H. (2010). Extracting representative tags for Flickr users. In *Proceedings of 10th international conference on data mining workshops* (pp. 312–317). Sydney: IEEE Press.

Flickr (2000). Flickr. http://www.flickr.com. Accessed 21 May 2011.

F-score (2002). Wikipedia F-score. http://en.wikipedia.org/wiki/F1_score. Accessed 15 Jan 2012.

Garg, N., & Weber, I. (2008). Personalized tag suggestion for Flickr. In *Proceedings of 17th International World Wide Web Conference* (pp. 1063–1064). Beijing: ACM Press.

Giannakidou, E., Koutsonikola, V., Vakali, A., & Kompatsiaris, I. (2011). In & out zooming on time-aware user/tag clusters. *Journal of Intelligent Information Systems, 37*, 1–24.

Heymann, P., Ramage, D., & Garcia-Molina, H. (2008). Social tag prediction. In *Proceedings of 31st annual international SIGIR conference on research and development in information retrieval* (pp.531–538). Singapore: ACM Press.

Julita, S., Sihem, A. Y., Cameron, M., & Cong, Y. (2008). Leveraging tagging to model user interests in del.icio.us. In *Proceedings of AAAI spring symposium on social information processing*. California: AAAI Press.

Liu, D., Hua, X. S., & Yang, L. (2009). Tag ranking. In *Proceedings of 18th world wide web conference* (pp. 351–360). Madrid: ACM Press.

Lu, Y., Yu, S., Chang, T. C., & Hsu, J. (2009). A content-based method to enhance tag recommendation. In *Proceedings of 21st international jont conference on artifical intelligence* (pp. 2064–2069). San Francisco.

Phan, N., Hoang, V., & Shin, H. (2010). Adaptive combination of tag and link-based user similarity in flickr. In *Proceedings of 10th international conference on multimedia* (pp. 675–678). Firenze: ACM Press.

Sen, S., Vig, J., & Riedl, J. (2009a). Learning to recognize valuable tags. In *Proceedings of 13th international conference on intelligent user interfaces* (pp. 87–96). Florida: ACM Press.

Sen, S., Vig, J., & Riedl, J. (2009b). Tagommenders: Connecting users to items through tags. In *Proceedings of 18th international world wide web conference* (pp. 671–680). Madrid: ACM Press.

Shin, H., Lee, J., & Hwang, K. (2010). Separating the reputation and the sociability of online community users. In *Proceedings of 25th ACM symposium on applied computing* (pp. 1807–1814). Switzerland: ACM Press.

Shin, H., Xu, Z., & Kim, E. (2008). Discovering and browsing of power users by social relationship analysis in large-scale online communities. In *Proceedings of 8th IEEE/WIC/ACM international conference on web intelligence* (pp. 105–111). Sydney: IEEE Press

Sigurbjörnsson, B., & Zwol, R. V. (2008). Flickr tag recommendation based on collective knowledge. In *Proceedings of 17th international world wide web conference* (pp. 327–336). Beijing: ACM Press.

Song, Y., Zhuang, Z., Li, H., Zhao, Q., Li, J., Lee, W., & Gile, C. L. (2008). Real-time automatic tag recommendation. In *Proceedings of 31st annual international SIGIR conference on research and development in information retrieval* (pp. 515–522). Singapore: ACM Press.

Suchanek, F. M., Vojnovi'c, M., & Gunawardena, D. (2008). Social tags: Meaning and suggestions. In *Proceedings of 17th conference on information and knowledge management* (pp.617–627). Napa Valley: ACM Press.

Weka (2001). The Website for Weka. http://www.cs.waikato.ac.nz/ml/weka/. Accessed 20 March 2011.

Witten, I. H., Paynter, G. W., Frank, E., Gutwin, C., & Nevill-Manning, C. G. (1999). KEA: Practical automatic keyphrase extraction. In *Proceedings of 4th ACM conference on digital libraries* (pp. 254–255). New Jersy: ACM Press.

Wu, L., Yang, L., & Yu, N. (2009). Learning to tag. In *Proceedings of 18th international world wide web conference* (pp. 361–370). Madrid: ACM Press.

Wu, X., Zhang, L., & Yu, Y. (2006). Exploring social annotations for the semantic web. In *Proceedings of 15th international world wide web conference* (pp. 417–426). Edinburgh.

Yih, W., Goodman, J., & Carvalho, V. R. (2006). Finding advertising keywords on web pages. In *Proceedings of 15th international world wide web conference* (pp. 213–222). Edinburgh: ACM Press.

Zhang, X., Li, Z., & Chao, W. (2011). Tagging image by merging multiple features in a integrated manner. *Journal of Intelligent Information Systems, 37*, 1–21.