Expert Systems with Applications 38 (2011) 8488–8496

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Collaborative user modeling with user-generated tags for social recommender systems

Heung-Nam Kim [a,*], Abdulmajeed Alkhaldi [a], Abdulmotaleb El Saddik [a,c], Geun-Sik Jo [b]

[a] School of Information Technology and Engineering, University of Ottawa, 800 King Edward, Ottawa, Ontario, Canada K1N 6N5
[b] School of Computer and Information Engineering, Inha University, 253 Younghyun-dong, Nam-gu, Incheon 402-751, Republic of Korea
[c] College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia

ARTICLE INFO

Keywords:
User modeling
Personalization
Recommender systems
Social tagging
Social media filtering

ABSTRACT

With the popularity of social media services, the sheer amount of content is increasing exponentially on the Social Web that leads to attract considerable attention to recommender systems. Recommender systems provide users with recommendations of items suited to their needs. To provide proper recommendations to users, recommender systems require an accurate user model that can reflect a user's characteristics, preferences and needs. In this study, by leveraging user-generated tags as preference indicators, we propose a new collaborative approach to user modeling that can be exploited to recommender systems. Our approach first discovers relevant and irrelevant topics for users, and then enriches an individual user model with collaboration from other similar users. In order to evaluate the performance of our model, we compare experimental results with a user model based on collaborative filtering approaches and a vector space model. The experimental results have shown the proposed model provides a better representation in user interests and achieves better recommendation results in terms of accuracy and ranking.

1. Introduction

Social media has been changing the way people find information, share knowledge and communicate with each other. This social phenomenon has transformed the masses, who were only information consumers via mass media, to be producers of information. However, as rich information is shared through social media sites, the huge amount of information that has not previously been available is increasing exponentially with daily additions. In turn, it is becoming increasingly more difficult for users to find the most attractive content and users often struggle with a great challenge in terms of information overload (Siersdorfer & Sizov, 2009).

Recommender systems that have emerged in response to the challenge provide users with recommendations of items that they would like the most (Adomavicius & Tuzhilin, 2005). To provide proper recommendations to users, the systems require information that includes a user's characteristics, preferences, and needs, typically referred to as a User Model (Godoy & Amandi, 2005). Therefore, building an accurate model for users is crucial to the success of recommender systems. One of the most successful technologies among recommender systems is Collaborative Filtering (CF) that makes the best use of "word-of-mouth" recommendations (Breese, Heckerman, & Kadie, 1998; Resnick, Iacovou, Suchak, Bergstorm, & Riedl, 1994). Although the field of CF research has a large number of information filtering problems, generally a typical CF domain starts with rating information that maps user-item pairs on a set of numerical values. Thus, user models in CF can be represented by ratings given by users on a set of items. Besides ratings, a number of modern services also allow the users to add tags to the items, known as social tagging. Consequently, a user model can profit by those tags in addition to ratings. Therefore, in the last few years, a number of studies have tried to combine recommender systems with social tagging in a way that can be highly beneficial to both areas (Milicevic, Nanopoulos, & Ivanovic, 2010). Such systems generate automated recommendations just as traditional recommender systems, but retain the flexibility of tagging information (Sen, Vig, & Riedl, 2009). In point of combining CF with tagging, they can be regarded as a new type of hybrid recommender systems that utilize human perceptive content contained in items. The reason is that a set of aggregated tags on an item is rich and compact enough to characterize and describe the same main concepts of the item although tag usage of users depends on a type of media items (e.g., articles, music, videos, and photos) they annotate (Bischoff, Firan, Nejdl, & Paiu, 2008; Li, Guo, & Zhao, 2008; Wetzker, Zimmermann, Bauckhage, & Albayrak, 2010).

In this study, we introduce a new method of building a user model that can represent a user's diverse preferences, and thus

* Corresponding author. Tel.: +1 613 562 5800x6248; fax: +1 613 562 5664.
E-mail address: hnkim@mcrlab.uottawa.ca (H.-N. Kim).

0957-4174/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2011.01.048

can be exploited to recommender systems. To build an individual user model, we connect tags and ratings as a way to infer a user's topics of interest, in which each topic is composed of tags. In addition, to provide the model with more diversity, valuable topics in terms of both likes and dislikes are enriched in collaboration with other similar users. For recommending items relevant to user needs, we seamlessly incorporate collaborative characteristics into a content-based filtering approach so that recommender systems exploit the benefits of each, and thus alleviate the cold start problem and the overspecialization issue. The cold start problem describes a new user joins a recommender system and has presented few opinions (e.g., rating and tagging). With these situations, the system is generally unable to make high quality recommendations (Schein, Popescul, Ungar, & Pennock, 2002). The cold start users should be encouraged to continuously provide their opinions because they do not have enough historical information. However, inaccurate recommendations from a dearth of reliable information on the users lead them to undermine the credibility of the system, and thus may cause their deviation from the system. Considering this point, a differentiated strategy of building the user model is necessary to make compelling recommendations for those users. On the other hand, overspecialization refers to situations where a user model exclusively relies on tags which are used by a user or labeled in his/her rated items. With those cases, it is hard to recommend novel items different from anything the user has previously rated (Adomavicius & Tuzhilin, 2005).

The main contributions of this paper toward user modeling in recommender systems can be summarized as follows: (i) We propose a new method of building a collaborative user model by leveraging user-generated tags. And we present a detailed method of topic-driven enrichments in collaboration with similar users that makes an individual model abundant. (ii) We present how the collaborative model can be applied to a recommender system. A locally weighted naïve Bayes approach is employed to recommend a ranked list of items relevant to users' needs. We show how well our approach effectively works in terms of improving the recommendation quality and in dealing with cold start users. (iii) We present a new approach to identify two sets of similar neighbors by seamlessly combining rating information and tagging information. In particular, we separate out the neighborhood with respect to relevant tags from the neighborhood with respect to irrelevant tags. We demonstrate how two separated neighborhoods offer an advantage over a single set of the neighborhood.

The rest of this paper is organized as follows: in next section we provide recent studies applying social tagging to recommender systems. In Section 3, we present a collaborative approach to user modeling for enriching valuable tags. We then provide a detailed description of how the proposed model is used for item recommendations in Section 4. In Section 5, we present the performance of our approach through experimental evaluations. Finally, conclusions are presented and future work is discussed in Section 6.

2. Related work

The popularity of the usage of user-generated tags allows us to capture valuable information for understanding user interests and thus to build better user models. There are several studies that exploit various aspects of user-generated tags to user modeling. Carmagnola et al. (2008) proposed a new approach of constructing an enriched user model by analyzing users' tagging behaviors and the meaning of tags. By applying the user model to a cultural heritage system, namely *iCITY*, the system supports personalized ways of navigating and tagging content, as well as recommend content to users. Li et al. (2008) presented a method of discovering common interests of users from user-generated tags. Analogous to our study, the authors discovered a set of patterns, i.e., topics of interest, that frequently co-occurred in tags added by users, and thus clustered users and resources according to each topic discovered. A similar approach is presented by Au Yeung, Cibbins, and Shadbolt (2009) as well. The authors constructed a user's model in the form of a set of frequent tag patterns representing multiple interests of the user. The concept of discovering frequent tag patterns and subsequently grouping users according to tags in the patterns is close to our work. However, differing from these two studies, our work identifies and enriched further tag patterns of value to each user in collaboration with other similar users. More recently, in Wang, Clements, Wang, De Vries, and Reinders (2010), three types of personalization models in collaborative tagging systems are introduced according to user tasks: a collaborative tagging model, a collaborative browsing model, and a collaborative item search model. Guy, Zwerdling, Ronen, Carmel, and Uziel (2010) aggregated relationship information among users, tags, and items across various social media services within enterprise application environment. From those aggregated relationships, a user model is represented that contains a set of related users and a set of related tags. Wetzker et al. (2010) proposed a user-centric tag model mapping individual tag vocabularies, known as personomies, on the corresponding folksonomies. They also presented how the model can be applied to tag recommendations and tag-based social search, rather than item recommendations.

In recent years, social tagging has also attracted considerable attention to recommender systems. In fact, many researchers have proposed a new application for recommender systems supporting the suggestion of suitable tags in annotating items. However, since our study focus on the collaborative nature of user modeling with social tagging for item recommendations, we mainly review studies dealing with item recommendations. Some early work in using tags for recommender systems is presented by Ji, Yeon, Kim, and Jo (2007). The authors first determine similarities between users with user-generated tags and subsequently identify the latent tags for each user by using a CF approach. Tso-Sutter, Marinho, and Thieme (2008) proposed a generic method that allows tags to be incorporated into standard CF algorithms by reducing three-dimensional correlations (i.e., user-item-tag) to three two-dimensional correlations and then applying a fusion method to re-associate these correlations. In Nakamoto et al. (2008), Reasonable Tag-based CF (RCF) is proposed assuming that tags generated by a user are synonymous with the reasons why the user liked an item. In RCF, tags are first clustered into topics by using an expectation-maximization (EM) algorithm. Afterwards, feature vectors of topics and items for each user, referred to as topic domain vectors, are created to find similar users and thus to recommend items in terms of topic domains. A Similar notion is previously described in De Gemmis, Lops, Semeraro, and Basile (2008) in which a user profile consists of two parts: *profile_like* and *profile_dislike*. The former contains features that help in finding relevant items whereas the latter helps in filtering out non-relevant items. A multivariate Poisson model for naïve Bayes is employed to estimate the posteriori probability and subsequently recommend a ranked list of items. There is the main difference between De Gemmis et al. (2008) and our work. Our work incorporates collaborative characteristics into a content-based recommendation that utilizes rating information and tagging information. On the other hand, De Gemmis et al. (2008) studied a typical content-based system that analyzes textual descriptions, such as summaries, reviews, and abstracts, in addition to ratings and user-generated tags. More recently, Siersdorfer and Sizov (2009) represented social tagging in the form of a vector space model that could apply to existing recommendation methods, i.e., content-based filtering and collaborative filtering. Sen and Vig (2009) introduced *Tagommenders* that predict users' preference for items by inferring the users' preference for tags. The authors extended content-based recommendation methods in a way that learned relationships between tags and items based on inferred tag preferences and item ratings. In Zhen, Li, and Yeung (2009), *TagiCoFi* is proposed integrating tagging information into a model-based CF, particularly using *relation regularized matrix factorization*. Liang, Xu, Li, Nayak, and Tao (2010) studied a user-based and an item-based CF combined with weighted tags, respectively. The authors considered all possible relationships between users, items and tags (i.e., user-item, tag-item and user-item similarities).

Similar to our motivation, the salient concept behind the above-mentioned studies is that social tagging can be highly beneficial to enhance the quality of recommender systems. Although the above-mentioned studies give reasonable promise of improving the performance, our study is different from earlier work. Because users often use personal and self-reference tags, mostly helpful for themselves, we look into what a tagged item is about and how much a user prefers the item, rather than capturing what tags are used by the user. To this end, in our study, we seamlessly integrate ratings as explicit user interests and tags as implicit user interests. We then discover frequent topics and tags existing in a user's set of both relevant and non-relevant items. In addition, rather than clustering users according to the topics, we carry out topic-driven enrichments of a user model with collaboration from other users so that the enriched model can reflect diverse topics not only relevant to user interests, but also irrelevant to user needs.

3. Collaborative user modeling

In this section, we describe our approach to building a user model that is derived from the user's ratings, as well as user-generated tags. Before going into further detail, the notation and definitions required for understanding our approach are introduced. Let $U = u_1, u_2, \ldots, u_{|U|}$ be the set of all users, $I = i_1, i_2, \ldots, i_{|I|}$ be the set of all items, and $T = t_1, t_2, \ldots, t_{|T|}$ be the set of all tags annotated in items. Let R be a user-item rating matrix in which $R_{u,i}$ represents the rating of user $u \in U$ on item $i \in I$. An element $R_{u,i}$ either exists as numerical ordinal scale or is unknown. We denote unknown ratings by $\emptyset$. The matrix $\mathbf{R}$ can be decomposed into row vectors: $\mathbf{R} = [\vec{r}_1, \ldots, \vec{r}_{|U|}]^T$ with $\vec{r}_u = [R_{u,1}, \ldots, R_{u,|I|}]$, for every $u \in U$. Therefore, each row vector represents the ratings of a particular user on items. We also denote a set of items rated by a certain user u as $I_u = i \in I | \forall i : R_{u,i} \neq \emptyset$. As for social tagging, a three-dimensional space between users, items, and tags (i.e., users assign tags to items) is projected onto a two-dimensional one. In this paper, we only take tag-item relationships into account, informing how many times a tag has been annotated in an item. We transform a bag-model of social tagging into a set-model indicating which tags occur and do not occur in a particular item. Formally, let Q be a tag-item binary matrix in which $Q_{t,i}$ is set to 1 if item i has been annotated at least l times or more with tag t and 0 otherwise.

3.1. Personal user model from tags

In general, ratings of a user for items reflect how relevant or interesting the items are to him/her. In addition, user-generated tags of an item are concise to the users understanding, and hence, capture content of the item (Li et al., 2008). It is worth examining how we can connect the user's ratings and the tags assigned by other users in respect to the item.

3.1.1. Positive and negative items

Although the rating scale is fixed as numerical values (e.g., a scale of 1–5), each user would have his/her own rating behavior. As shown in Fig. 1, from the ratings of items that have previously rated by a user, we classify the items into two portions: a set of positive (relevant) items and a set of negative (irrelevant) items. If a rating of a user for a certain item is greater than or equal to the average rating $\overline{R_u}$ of the user, we then call this item a positive item for him/her and a negative item otherwise. More formally, the set of positive and negative items for user u are denoted as $Pos(u)$ and $Neg(u)$, respectively such that $Pos(u) = i \in I | R_{u,i} \geqslant \overline{R_u}$, $Neg(u) = i \in I | R_{u,i} < \overline{R_u}$, and $Pos(u) \cap Neg(u) = \emptyset$.

3.1.2. Calculating weights of tags

In our study, we associate a weight of tags with a user's rating, rather than the well-known *tf-idf* weight. The calculation of the weight for tags is straightforward. First, the vector of ratings $\vec{r}$ for each user is normalized as $\|\vec{r}\| = 1$, and subsequently, the normalized values are used for weights in terms of a user for a tag. Formally, the weight of tag t annotated in item i for user u, denoted as $w_{u,i}(t)$, is computed by:

$$w_{u,i}(t) = \frac{R_{u,i}}{\sqrt{\sum_{j=1}^{|I|} R_{u,j}^2}} \quad (1)$$

Because a tag may appear in multiple items with different weights that quantify the interest of the tag for a certain user, we compute the mean weight of the tag in the set of positive items and negative items, respectively:

$$\mu_{u,t}^{pos} = \frac{1}{|I_u^{pos}(t)|} \times \sum_{j \in I_u^{pos}(t)} w_{u,j}(t), \quad \mu_{u,t}^{neg} = \frac{1}{|I_u^{neg}(t)|} \times \sum_{j \in I_u^{pos}(t)} w_{u,j}(t) \quad (2)$$

where $I_u^{pos}(t)$ is the set of positive items rated by user u containing tag t and $I_u^{neg}(t)$ is the set of negative items rated by user u containing tag t. Finally, the global weight of tag t for user u, denoted as $\mu_{u,t}$, can be illustrated by the following equation:

$$\mu_{u,t} = \begin{cases} (\mu_{u,t}^{pos} + \mu_{u,t}^{neg})/2, & \text{if } t \in I_u^{pos}(t), t \in I_u^{neg}(t) \\ \mu_{u,t}^{pos}, & \text{if } t \in I_u^{pos}(t), t \notin I_u^{neg}(t) \\ \mu_{u,t}^{neg}, & \text{if } t \in I_u^{neg}(t), t \notin I_u^{pos}(t) \end{cases} \quad (3)$$

3.1.3. Discovering topics

We discover frequent tag patterns from a set of items rated by each user. Because every user has different tastes on items, a dataset used for the mining process should also be selected individually for each user. In the data mining research literature, frequent patterns are typically defined as patterns that occur at least as frequently as a predetermined threshold, commonly referred to as a *minimum support* (Han, Pei, & Yin, 2004). In our study, frequent patterns are a set of tags that appear frequently together in either a set of a user's positive items $Pos(u)$ or a set of a user's negative items $Neg(u)$.

As observed by previous studies (Bischoff et al., 2008; Li et al., 2008), user-generated tags can cover the main concepts of an item and provide a higher-level abstraction on content of an item. Furthermore, according to a critical characteristic of social tagging, also known as *folksonomy*, vocabularies that emerge organically from the tags can be considered to describe a particular topic for the item. In this sense, if multiple tags are frequently annotated together with each other in positive items of a user, the user would be interested in a particular topic described by the co-occurred tags. On the other hand, a set of tags appear frequently together in a user's set of negative items, the user might dislike a topic containing the tags. Such set of tags, i.e., a frequent tag pattern, can be discovered using existing frequent-pattern mining methods such as *Apriori* (Agrawal & Srikant, 1994) and *FP-growth* (Han et al., 2004). Since a set of items rated by a user is divided into a set of positive items and a set of negative items, frequent tag patterns can also be divided into two por-

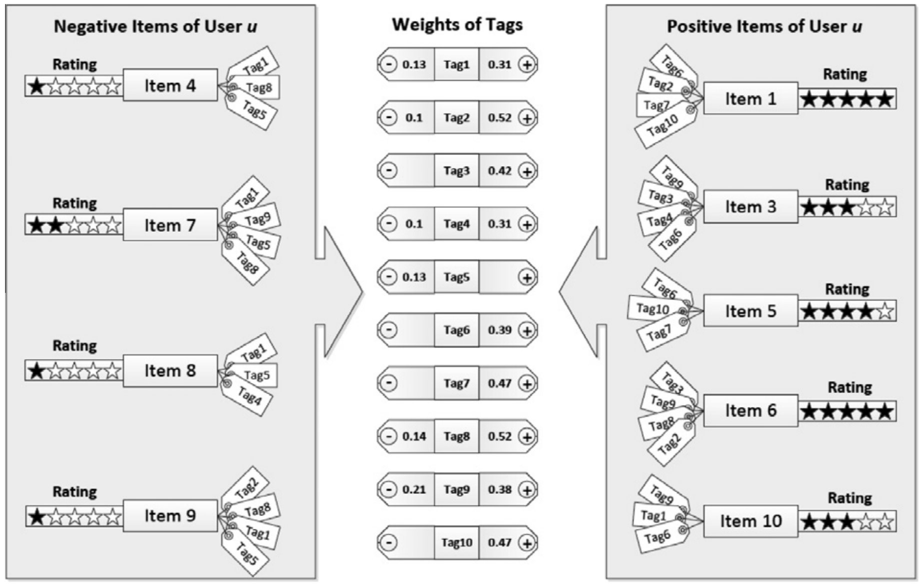

Fig. 1. Classification of positive items and negative items for a user according to his/her item ratings.

tions according to the sets used for the mining process. We regard patterns discovered from the positive items as implicitly *relevant topics*, and patterns discovered from the negative items as implicitly *irrelevant topics*, defined as follows:

Definition 1 (*Relevant topics*). Let pattern $p = t_1, t_2, \ldots, t_{|p|}$ be a set of tags such that $p \subseteq T$. A relevant topic is defined as tag pattern p for which *support* of pattern p (i.e., the ratio of items in the set $Pos(u)$ that contains pattern p, written as $Sup_u(p)$) is greater than or equal to the minimum support for user u, θ_u.

Definition 2 (*Irrelevant topics*). Analogous to a relevant topic, a irrelevant topic is defined as a tag pattern for which the ratio of items in the set $Neg(u)$ that contains the pattern is greater than or equal to the minimum support for user u, θ_u.

According to the downward closure property of frequent patterns, if a tag pattern is a frequent pattern, then all subsets of that pattern is also frequent patterns (Agrawal & Srikant, 1994). As pointed out in Li et al. (2008), if all subsets of a certain pattern have the same support value, then they are always annotated in the same items among positive items (or negative items) of a user. In this case, the specific pattern may be more valuable to the user than each individual pattern derived from the specific one. Inspired by Li et al. (2008), we prune such subset patterns whose support values are the same as one another. We will henceforth use the term '*topic*' to refer to the frequent tag patterns. Formally, for a given user u, we denote by ToR_u and ToI_u the set of relevant topics and irrelevant topics, respectively. We present a formal description of a model for user u as follows: $IM_u = \langle ToR_u, ToI_u \rangle$, which is referred to as an *initial model* for him/her. To facilitate the following sections, we further define some notations here. A tag occurring within the set ToR_u is called a *positive tag* and a set of positive tags for user u is denoted as IT_u^{pos}. Analogously, a tag occurring within the set ToI_u is called a *negative tag* and a set of negative tags for user u is denoted as IT_u^{neg}.

3.2. Identifying tag-based neighborhood

Before enriching a model for a certain user, we first need to form neighborhood of the user. The main goal of neighborhood formation is to identify a set of users similar to a particular user, commonly referred to as *neighbors*. A typical collaborative filtering encounters serious limitations for finding a set of neighbors. In practice, even when users are very active, the result of rated items is only a small proportion of the total number of items. Accordingly, it is often the case that a pair of users has nothing in common, and hence the similarity cannot be computed (Sarwar, Karypis, Konstan, & Reidl, 2001). Even when the computation of similarity is possible, it may not be very reliable, because insufficient information is processed. To that end, recent studies have determined similarities between users by using user-generated tags instead of user ratings (Ji et al., 2007; Liang et al., 2010; Markines et al., 2009; Nakamoto et al., 2008; Siersdorfer & Sizov, 2009; Zhen et al., 2009). In this paper, we also identify the best neighbors based on tags labeled in items. However, differing from the previous work, rating information is embedded into the tags when we compute the similarities between users, rather than frequency-based weights for the tags. Moreover, as two users could like some topics in common but could have a difference with respect to dislikes, neighbors in terms of relevant topics are maintained to be separated from neighbors in terms of irrelevant ones.

In order to find k similar neighbors, the cosine similarity, which quantifies the similarity of a pair of vectors according to their angle, is employed to measure the similarity values between a target user and every other user. Because a model of each user consists of both relevant topics and irrelevant topics, we compute two similarity values. Formally, the similarity between a pair of users, u and v in terms of relevant topics is measured by Eq. (4)

$$sim_{u,v}^{pos} = \frac{\sum_{t \in (IT_u^{pos} \cap IT_v^{pos})} \mu_{u,v}^{pos} \times \mu_{u,t}^{pos}}{\sqrt{\sum_{t \in IT_u^{pos}} \mu_{u,t}^{pos^2}} \times \sqrt{\sum_{t \in IT_u^{pos}} \mu_{v,t}^{pos^2}}} \quad (4)$$

where IT_u^{pos} and IT_v^{pos} refer to the set of tags in relevant topics of user u and v, respectively. Analogously, the similarity between user u and v with respect to irrelevant topics can also be calculated as:

$$sim_{u,v}^{neg} = \frac{\sum_{t \in (IT_u^{neg} \cap IT_v^{neg})} \mu_{u,v}^{neg} \times \mu_{u,t}^{neg}}{\sqrt{\sum_{t \in IT_u^{neg}} \mu_{u,t}^{neg^2}} \times \sqrt{\sum_{t \in IT_u^{neg}} \mu_{v,t}^{neg^2}}} \quad (5)$$

where IT_u^{neg} and IT_v^{neg} refer to the set of tags in irrelevant topics of user u and v, respectively.

The similarity value between a pair of users is in the range [0, 1] and the higher a user's value, the more similar he/she is to a target user. Finally, for a given user $u \in U$, particular k users with the highest similarity are identified as *positive neighbors* and *negative neighbors* such that:

$$N_k^{pos}(u) = \underset{v \in U \setminus \{u\}}{\arg\overset{k}{\max}}\, sim_{u,v}^{pos},$$
$$N_k^{neg}(u) = \underset{v \in U \setminus \{u\}}{\arg\overset{k}{\max}}\, sim_{u,v}^{neg} \quad (6)$$

3.3. Enriching a user model

Once we have identified the set of the nearest neighbors for a certain user u, his/her initial model $IM_u = \langle ToR_u, ToI_u \rangle$ described in Section 3.1 is enriched from the neighbors. The basic idea of enriching the model starts from assuming that the user is likely to prefer similar tag patterns discovered within his/her neighbors. For example, if positive items of a user frequently contain tags "web2.0" and "semanticweb," he/she may also be interested in topics, such as "web2.0," "semanticweb," and "ontology," that frequently appears in positive items of users similar to him/her. This enrichment process is particularly effective to some users who do not contain topics sufficiently in their user model.

As illustrated in Fig. 2, we elaborate on the general idea of topic-driven enrichments in the following example: Firstly, we choose neighbor user $\vee \in N_k^{pos}(u)$ in descending order of similarity between target user u and neighbors. For each topic p_i in ToR_u, *specific topics* of p_i in ToR_v are identified. Given two topics $p_i \in ToR_u$ and $p_j \in ToR_v$, p_i is said to a *general topic* of p_j if and only if p_i is a subset of p_j, i.e., $p_i \subset p_j$. On the contrary, p_j is called a *specific topic* of p_i. For example, let $p_1 = \{t_1, t_2\}$ be a relevant topic of user u such that $p_1 \in ToR_u$, and $ToR_v = \{p_2, p_3, p_4, p_5\}$ be the set of relevant topics of user v such that $p_2 = \{t_1, t_2, t_3\}$, $p_3 = \{t_1, t_2, t_4\}$, $p_4 = \{t_1, t_2, t_3, t_5\}$, and $p_5 = \{t_3, t_4\}$. Since topic p_2, p_3, and p_4 contain the entire tags of topic p_1, they are said to the specific topics of p_1 whereas p_5 is not the specific topic. In other words, p_1 is said to the general topic of topics p_2, p_3, and p_4. Several specific topics occurring in the model of neighbor v may be found. For efficient enrichment, we only consider specific topics that have a higher support than that of a general topic. For example, assume that the support for p_1, p_2, p_3, and p_4 is 0.41, 0.5, 0.47, and 0.35, respectively (i.e., $Sup_u(p_1) = 0.41$, $Sup_v(p_2) = 0.5$, $Sup_v(p_3) = 0.47$, and $Sup_v(p_4) = 0.35$). In this case, only topic p_2 and p_3 are used for enriching the model of user u if they are not included in the set ToR_u. Topics such as p_2 and p_3 are called *enriched topics* for user u, and tags such as $t_3 \in p_2$ and $t_4 \in p_3$ are called *enriched tags* for user u.

Finally, a set of enriched topics is identified from k *positive neighbors* of target user u. Note that some topics that were previously enriched by a certain neighbor v are also identified from another neighbor h such that $sim_{u,v}^{pos} \geqslant sim_{u,h}^{pos}$, for $v \neq h$. If a particular topic is enriched many times from the neighbors, the target user is likely to be more interested in that topic. In an analogous fashion, with respect to irrelevant topics, a set of enriched topics can be determined from k *negative neighbors* $N_k^{neg}(u)$ of user u. Formally, the enriched model, for a given user u, is defined as a tuple $EM_u = \langle EToR_u, EToI_u \rangle$ where $EToR_u$ is the set of relevant topics enriched from positive neighbors such that $EToRu \cap ToRu = \varnothing$, and $EToI_u$ is the set of irrelevant topics enriched from negative neighbors such that $EToI_u \cap ToI_u = \varnothing$. For convenience, we also define any enriched tags occurring within either $EToR_u$ or $EToI_u$ as the sets: a set of enriched tags from *positive neighbors* ET_u^{pos} such that $ET_u^{pos} \cap IT_u^{pos} = \varnothing$, and a set of enriched tags from *negative neighbors* ET_u^{neg} such that $ET_u^{neg} \cap IT_u^{neg} = \varnothing$. Eventually, we represent user u with the initial model IM_u and the enriched model EM_u. We label the unified model the *collaborative model* for user u: $CM_u = \langle IM_u, EM_u \rangle$.

4. Recommendation via probabilistic approach

The final step is to generate recommendations such as a list of top-N items that a user would like the most. In our study, an item recommendation is treated as a classification problem that an item belongs to either a positive class *Pos* or a negative class *Neg* depending on a *collaborative model* CM_u. To this end, we adopt *locally weighted naïve Bayes* (Frank, Hall, & Pfahringer, 2003) with the multinomial event model (McCallum & Nigam, 1998) to recommend top-N items stochastically. To apply the *naïve Bayes* classifier to the recommendation process of the target user u, for a given item $i \notin I_u$, tags annotated in item i such that $T_i = t_j \in T | \forall : Q_{j,i} = 1$ are used as feature variables. More formally, *naïve Bayes* model for computing a posterior probability can be written as:

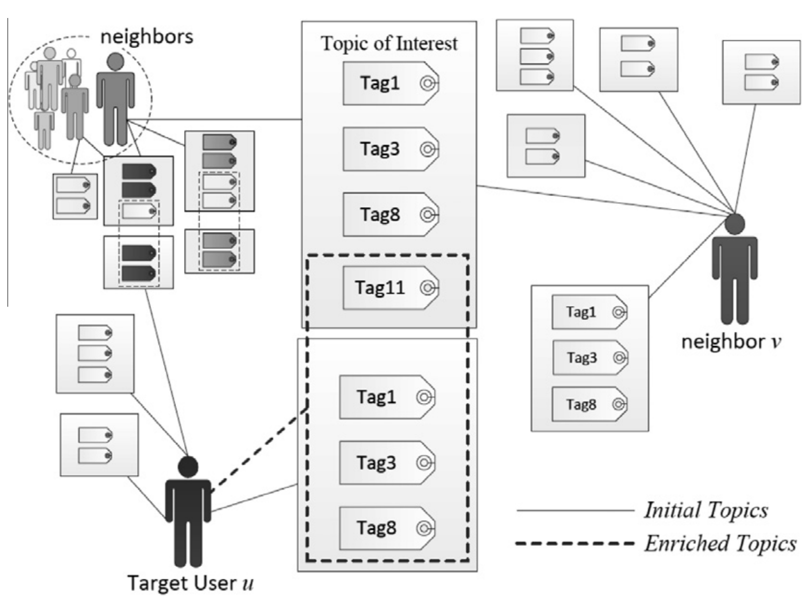

Fig. 2. Enriching topics of a target user from similar neighbors.

$$P(Pos|i) = \frac{P(Pos)}{P(i)} \prod_{t_j \in T_i} P(t_j|Pos), \quad P(Neg|i) = \frac{P(Neg)}{P(i)} \prod_{t_j \in T_i} P(t_j|Neg) \tag{7}$$

If we take the logarithm after dividing one by the other for a log-likelihood ratio, then we can use the following equation:

$$z_{u,i} = (\ln P(Pos) - \ln P(Neg)) + \sum_{j=1}^{|T_i|} (\ln P(t_j|Pos) - \ln P(t_j|Neg)) \tag{8}$$

where the prior probability for the class *Pos* and *Neg* are computed as $P(Pos) = |Pos(u)|/|I_u|$ and $P(Neg) = |Neg(u)|/|I_u|$, respectively. In addition, locally weighted conditional probabilities $P(t_j|Pos)$ and $P(t_j|Neg)$ with Laplacian smoothing are estimated as follows:

$$\begin{aligned} P(t_j|Pos) &= \frac{1 + \omega_j \cdot f_{u,j}^{pos}}{V_u + \sum_{t \in (IT_u^{pos} \cup ET_u^{pos})} \omega_t \cdot f_{u,t}^{pos}} \\ P(t_j|Neg) &= \frac{1 + \omega_j \cdot f_{u,j}^{neg}}{V_u + \sum_{t \in (IT_u^{neg} \cup ET_u^{neg})} \omega_t \cdot f_{u,t}^{neg}} \end{aligned} \tag{9}$$

where $f_{u,j}^{pos}$ and $f_{u,j}^{pos}$ are respectively the frequency of positive tag t_j and negative tag t_j in CM_u. V_u is the total number of distinct tags in CM_u and ω_j is the weight of tag *j* for user *u* given by:

$$\omega_j = \begin{cases} \mu_{u,j}, & \text{if } t_j \in IM_u \\ \mu_{v,j}, & \text{if } t_j \in EM_u, \ t_j \in IM_v \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

where user v is the most similar user who contributes to enrich tag *j* to user *u*. This allows us to put larger weights on more relevant tags and smaller weights on irrelevant ones.

Finally, a set of *N* ordered items that obtain the higher log-likelihood ratios is recommended to user *u* such that:

$$TopN(u) = \underset{i \in I \setminus I_u}{\arg\overset{N}{\max}}\, z_{u,i} \tag{11}$$

As discussed before, recommender systems relying exclusively on a user's rated items and his/her tags may only recommend items highly related to content the user has previously rated. It is hard to recommend novel items that are different from anything the user has previously rated, i.e., overspecialization problem (Adomavicius & Tuzhilin, 2005). In our model, by utilizing enriched topics from neighbors similar to a particular user, items containing enriched tags of great value to the user rank higher in the recommended set.

5. Experimental evaluation

5.1. Experimental setup

As the focus of this study is to build an effective model of users derived from rating information with tagging information, it was difficult to find publicly available datasets that plentifully contain both tagging and rating information at the same time. Therefore, we designed a Web-based interface that allows users to rate items with numerical values, as well as annotate them with tags. One hundred and thirty-three participants who are university students were invited to our experiment. To gather sufficient rating information from the participants, we decided to select top box-office movies[1] and top rated movies[2] in The Internet Movie Database (IMDb). After removing movies overlapped, 592 movies were extracted in total. Whenever the participants rated movies with 1-to-5 star scale, the participants were also encouraged to add tags to the movies as many as possible. For minimizing noise tags and providing users' convenience, candidate tags were suggested to the participants by utilizing plot keywords and genres of each movie presented by IMDb. It should be noticed that our study is more interested in capturing which tags are labeled in items rather than which tags are used by users. In total, we collected 10,570 ratings on 592 items from 133 users (i.e., 133 rows and 592 columns of a user-item matrix **R**). As for tagging data, we considered tags that were added to a particular movie by at least three more users because some tags were meaningless to other people except for users who create the tags. We collected 63,325 non-zero entries (i.e., tag assignments on items) of a tag-item binary matrix **Q** after pruning some noise tags.

The experiments have been designed to answer the following questions:

- Is identifying two separate sets of neighbors effective for enriching user models?
- Is the enrichment of user models effective for improving accuracy and ranking?
- Is the quality of recommendations based on the enriched model competitive against the existing approaches?
- Is the proposed approach able to provide proper recommendations even if users rated few items?

To evaluate the performance of the recommendations, we divided the dataset into *a training set* and *a test set*. For each user *u* in the dataset, we randomly selected 10 positive items, and subsequently used those as his/her test set. And the remaining items which the user previously rated were used as the training set. When building users' initial model from the training set, we set the minimum support θ_u for each user to 0.2 (20%). To ensure that our results are not sensitive to the particular training/test portioning for each user, the experiments were repeated five times with different the training/test set. Therefore, the result values reported in the experiment sections are the averages over all five runs.

5.2. Evaluation metrics

Error metrics such as Root Mean Square Error (RMSE) and Mean Absolute Error (MAE), which are commonly used metrics for evaluating prediction accuracy, do not directly measure the quality of *top-N* recommendations (Cremonesi, Koren, & Turrin, 2010; McLaughin & Herlocker, 2004). Therefore, we adopted two evaluation measures as follows:

Precision at top N. In the context of top *N* recommendations, precision can judge how relevant a set of ranked recommendations is for users. Precision measures the ratio of items in a list of recommendations that were also contained in the test set to number of items recommended (Herlocker, Konstan, Terveen, & Riedl, 2004). Note that precision in recommender systems is generally an underestimate of the true precision because we treat non-rated items as non-relevant items (McLaughin & Herlocker, 2004). Precision at top *N* for a given user u in test set is given:

$$P(u)@N = \frac{|Test(u) \cap TopN(u)|}{|TopN(u)|} \tag{12}$$

where *Test*(u) is the set of items rated by user *u* in the test data whereas *TopN*(u) is the set of top *N* items recommended to user *u*. Finally, the overall *precision* at top *N* for all users in the test set is computed by averaging the personal *precision*.

Ranking accuracy at top N. Although precision is an efficient measure, it does not consider an item's rank within a list of top *N* items recommended. That is, an item recommended with top ranking is treated equally with an item that is recommended with *N*th ranking. Therefore, we also used the ranking accuracy metric,

[1] http://www.imdb.com/boxoffice/alltimegross.
[2] http://www.imdb.com/chart/top.

which was introduced by Deshpande and Karypis (2004). The ranking accuracy of user u at top N is defined as:

$$RK(u)@N = \sum_{i \in Test(u) \cap TopN(u)} \frac{1}{rank(i)} \tag{13}$$

where $rank(i)$, $1 \leqslant rank(i) \leqslant N$, refers to the recommended rank of item i within the top N list of user u. That is, relevant items that appear earlier in the *top-N* list are given more weight than later ones. The higher the $RK(u)@N$ value, the more accurately the algorithm ranks items to user u. Finally, the overall raking accuracy for all users is computed by averaging the personal raking value in the test data.

5.3. Experiments with neighborhood size

The following experiment investigates the influence of neighbors on the enrichment of the user model. For this reason, different numbers of neighbors k were used for model enrichment: 10, 20, 30, 40, 50, and 60. In addition, we also adopted a standard similarity measurement, which employed the cosine similarity using the rating matrix $\mathbf{R}$, during the neighborhood formation of each user. In fact, different measurements lead to a different set of neighbors for a user, in turn, leading to different topics and tags enriched.

Table 1 and Table 2 summarize the results according to different numbers of neighbors. The rows labeled "*Pos*/*Neg*" show the results of the user model in collaboration with positive neighbors and negative neighbors described in Section 3.2. And the rows labeled "ratingSim" show the results of the collaborative model enriched from rating-based standard neighbors.

Comparing the results achieved by *Pos*/*Neg* and ratingSim, the enriched model which is derived from positive and negative neighbors separately is more accurate than the model formed from the single neighborhood. For example, when the neighborhood size k is 10, *Pos*/*Neg* achieves 3.26% improvement over ratingSim in terms of P@10. With respect to RK@10, similar results are demonstrated. Nevertheless, we observed that the model based on ratingSim also seemed to work well. That may be caused by rating density in the matrix $\mathbf{R}$. The density of the rating matrix we used was relatively high, and thus, we could identify reliable neighbors using rating information. Examining the value of each case in small sizes of the neighborhood, the both cases provide a reasonably good performance in comparison with large sizes of the neighborhood. This result shows that the accuracy of our model is relatively insensitive to the value of k. That is, once the number of nearest neighbors is relatively large enough, the rank of recommended items for each user is barely changed by any further increases in the number of neighbors even though the slight variation of the precision and ranking values appeared. In other words, the neighborhood with a small size provides enough to enrich topics for each user because the useful information has already been enriched well by more similar neighbors.

In consideration of both precision and raking accuracy, the neighborhood size for enriching the model of each user was set to 40 in subsequent experiments.

Table 1
Precision at top 10 with respect to increasing k value.

Neighbors (k)	10	20	30	40	50	60
Pos/*Neg*	0.2235	0.222	0.2210	0.2210	0.2247	0.2273
ratingSim	0.1909	0.2010	0.1909	0.1909	0.1970	0.1919

Table 2
Ranking accuracy at top 10 with respect to increasing k value.

Neighbors (k)	10	20	30	40	50	60
Pos/*Neg*	0.8249	0.8491	0.8484	0.8596	0.8447	0.8562
ratingSim	0.8172	0.8177	0.8172	0.8125	0.8266	0.8122

5.4. Effect of model enrichment

This section investigates the effect of the collaborative model CM_u of each user in more detail, by comparing the results obtained by the initial model IM_u of each user.

Fig. 3 shows the results of the experiment. The results demonstrate that the collaborative model enriched by neighbors enhances the performance of the initial model. With respect to precision, *CUM* achieves 3.6% improvement. More importantly, it is apparent that the collaborative model considerably improved the ranking values, compared to the initial model. This is particularly important because users tend to be concerned with items having higher ranks. The results confirm that the enriched tags and topics positively impact on the user model and consequently the collaborative model has advantages in terms of improving both the recommendation accuracy and its ranking.

5.5. Comparisons with other methods

In this section, we present detailed experimental results in comparison with baseline methods. For comparison purposes, we tested the following baselines: (i) a CF based on a user-to-user similarity *UTU* (Breese et al., 1998), (ii) a CF based on an item-to-item similarity *ITI* (Deshpande & Karypis, 2004), and (iii) a tf-idf vector space model using tags *VSM*. Our collaborative model *CUM* was then compared with the baseline algorithms in top-*N* recommendations. As pointed out in the previous studies (Cremonesi et al., 2010; McLaughin & Herlocker, 2004), directly using predicted ratings as ranking score may not accurately recommend items. In fact, we generated top-*N* recommendations with CF approaches based on predicted ratings. However, the performance on precision is remarkably low even though prediction accuracy was quite good. Therefore, CF methods are designed to compute ranking scores achieved by non-normalized weighted sum of ratings using the cosine similarity between two users (for *UTU*) or two items (for *ITI*) as the weight. We also conducted the parameter tuning experiment with *UTU* and *ITI* beforehand in order to choose the best neighborhood size. Consequently, the size of *UTU* and *ITI* was set to 80 and 50, respectively. As for *VSM*, it was designed to build a user model from tags in his/her positive and negative items. Simply, we subtracted tag weight vectors of the negative items from tag weight vectors of the positive items. Afterwards an item was ranked using the calculated cosine similarity between tags annotated in the item and the user model.

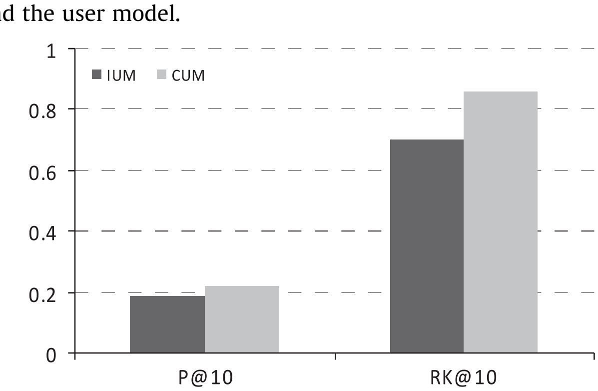

Fig. 3. Comparison of P@10 and RK@10 obtained by the initial model and the collaborative model.

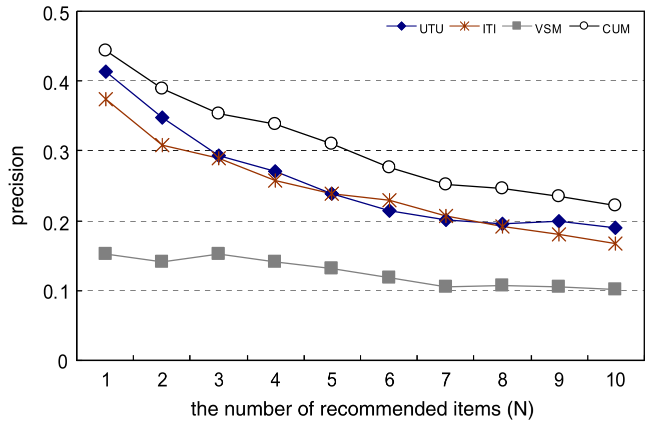

Fig. 4. Comparisons of *precision* as the number of recommended items *N* increases.

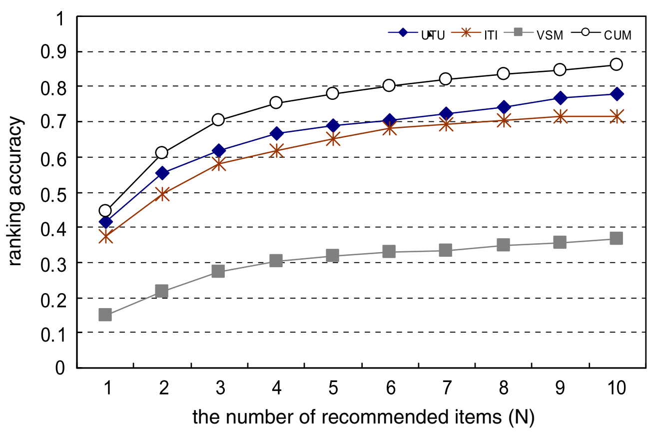

Fig. 5. Comparisons of ranking accuracy as the number of recommended items *N* increases.

To experimentally evaluate the performance of top-*N* recommendation, we calculated *P@N* and *RK@N* of each method no matter whether users have sufficient ratings or not. We selectively varied the number of returned items *N* from 1 to 10. First, we measured *P@N* obtained by *UTU*, *ITI*, *VSM*, and *CUM*.

Fig. 4 shows the results of precision showing how *CUM* outperforms the baseline methods. The graph curves show that the precision values tend to decrease as the number of recommended items *N* increases. However, in the case of *VSM*, there is very little difference for different values of *N*. Comparing the results achieved by *CUM* and the baseline methods, the precision value of the former was found to be superior to that of the benchmark methods in all cases. When compared to *VSM*, *CUM* is significantly more accurate on precision. On average, on all occasions, *CUM* outperforms *UTU*, *ITI* and *VSM* by 5%, 6.2% and 18.1%, respectively.

With respect to the ranking accuracy, Fig. 5 confirms that *CUM* continues outperforming the other methods on all variations of *N* value. For example, when *N* is 10, *CUM* obtains an *RK* value of 0.86 whereas *UTU*, *ITI*, and *VSM* demonstrates an *RK* value of 0.78, 0.71, and 0.37, respectively. The result can indicate that our model provides more suitable items with a higher rank in the recommended set than the other methods, and consequently makes compelling recommendations for all users. More interestingly, the simple vector model did not perform well, compared to the other methods. This result might be caused by the different degree of the density of matrices. In general, it is known that CF produces good performance in situation where a rating matrix is dense (Herlocker et al., 2004). As mentioned previously, the density of the user-item matrix **R** we used was 13.4% whereas that of the tag-item matrix **Q** was 3.8%.

We further examined the recommendation performance for users who had few ratings, namely cold start users, and had lots of ratings, namely active users, in the training set. Recommender systems are generally unable to make high quality recommendations, compared to the case of active users, which is pointed out as one of the limitations. We selectively considered two groups of users who have less than 10 ratings and greater than 250 ratings. And for two groups we calculated precision and ranking at top-10 (i.e., *P*@10 and *RK*@10) obtained by the four algorithms in order to analyze whether the differences between the two groups are statistically significant or not. Fig. 6 shows the results for the cold start group and active group.

As we can see from the graphs, the results of the baseline algorithms demonstrated that the values of the two metrics for the cold start group were considerably low. However, our method provided quite consistent performance no matter whether they have sufficient ratings or not. Such results were caused by the fact that the baseline methods were hard to analyze the users' propensity for items because they did not have enough information (ratings or tags). In contrast, our collaborative user model could represent suitable preference of the users by enriching both relevant topics and irrelevant topics from similar neighbors. Comparing the results achieved by *CUM* and the baseline algorithms, for the cold start users, precision and ranking values of the former was found to be superior to those of the other methods. For example, *CUM* obtains 10%, 15%, and 13% improvement for precision compared to *UTU*, *ITI*, and *VSM*, respectively. With respect to the ranking accuracy, it is clear that *CUM* significantly outperforms the three methods. This result indicates that our user model can help, indeed, in alleviating the problem of the cold start users and thus in improving the quality of item recommendations.

With respect to the active group, it can be observed that CF approaches (i.e., *UTU* and *ITI*) provide better performance than our method. The result indicates that collaborative filtering relatively works well when users have abundant rating information. Comparing results for the cold start group and the active group obtained by the CF approaches, it is apparent that the two groups have

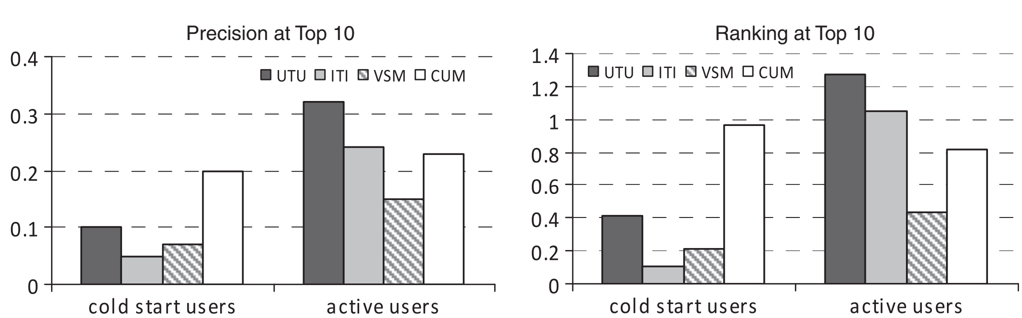

Fig. 6. Comparisons of *precision* and ranking accuracy for cold start users and active users.

significant performance disparity. For example, the active users based on *UTU* and *ITI* obtain 22% and 19% improvement in terms of *P*@10, compared to the cold start users, respectively. Similar levels of improvement can be seen on *RK*@10. This implies that the number of ratings is a significant factor affecting the quality of the recommendation in CF systems. However, in practice, even though users are very active, each individual has only expressed a rating on a very small portion of the total number of items. On the contrary to the CF approaches, there is very little difference in precision for the two groups in *CUM*, although it appears that the active users achieved slightly better results than the cold start users. Comparing results of *RK*@10 in the cold start and active group achieved by *CUM*, interesting results were observed. The value of the active group was rather worse than that of the cold start group. That is, enriched topics for recommendations works well for the cold start users. However, for the active users, enriched topics can give negative influence on performance because they may already have satisfactory relevant and irrelevant topics for themselves. Although the performance of *CUM* is worse than that of the CF methods in the active group, notably the proposed method produces very consistent quality of item recommendations for both the cold start users and the active users.

6. Conclusions and future work

For the future of the Social Web, social recommender systems with tags are becoming widely used as an important technique to enhance the quality of recommendations. In this paper, we have presented a method of building a user model incorporated with ratings and tags. We also propose the process of topic-driven enrichments to discover tags of value for users, in turn, enabling an individual model to be abundant. Empirically, the recommendation method based on our collaborative model outperforms the initial model, as well as the standard baseline methods. In addition, the proposed approach seems to produce rather robust performance in the situation where users have insufficient information about their preference, i.e., cold start users.

Although the approach presented in this study has shown promising results, it has also opened several tasks for interesting future work. First, as pointed out as a common problem in free-text tags, users make frequent use of ambiguous and synonymous tags (Golder & Huberman 2006). The recent systems, such as Faviki[3] and Zigtag,[4] support semantic tagging that allows users to assign tags with well-defined concepts. By considering semantics of tags, we intend to analyze further the inside of user-generated tags in order to build better a user model. Second, to deal with a different scenario, we intend to investigate the possible usages of our model for social search, which is one of emerging topics in the Social Web.

References

Adomavicius, G., & Tuzhilin, A. (2005). 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), 734–749.

Agrawal, R., & Srikant, R. (1994). Fast algorithms for mining association rules. In *Proceedings of 1994 international conference on very large data bases* (pp. 487–499).

Au Yeung, C. M., Cibbins, N., & Shadbolt, N., 2009. Multiple interests of users in collaborative tagging systems. In I. King & R. Baeza-Yates, (Eds.), *Weaving services and people on the World Wide Web* (pp. 255–274). Springer-Verlag.

Breese, J. S., Heckerman, D., & Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the 14th conference on uncertainty in artificial intelligence* (pp. 43–52).

[3] http://www.faviki.com.

[4] http://www.zigtag.com.

Bischoff, K., Firan, C. S., Nejdl, W., & Paiu, R. (2008). Can all tags be used for search. In *Proceeding of the 17th ACM conference on information and knowledge management* (pp. 203–212).

Carmagnola, F., Cena, F., Console, L., Cortassa, O., Gena, C., Goy, A., et al. (2008). Tag-based user modeling for social multi-device adaptive guides. *User Modeling and User-Adapted Interaction, 18*(5), 497–538.

Cremonesi, P., Koren, Y., & Turrin, R., 2010. Performance of recommender algorithms on top-*n* recommendation tasks. In *Proceedings of the 4th ACM conference on recommender systems* (pp. 39–46).

De Gemmis, M., Lops, P., Semeraro, G., & Basile, P. (2008). Integrating tags in a semantic content-based recommender. In *Proceedings of the 2nd ACM conference on recommender systems* (pp. 163–170).

Deshpande, M., & Karypis, G. (2004). Item-based top-*n* recommendation algorithms. *ACM Transactions on Information Systems, 22*(1), 143–177.

Frank, E., Hall, M., & Pfahringer, B. (2003). Locally weighted naive Bayes. In *Proceedings of the 19th conference on uncertainty in artificial intelligence* (pp. 249–256).

Godoy, D., & Amandi, A. (2005). User profiling in personal information agents: A survey. *The Knowledge Engineering Review, 20*(4), 329–361.

Golder, S. A., & Huberman, B. A. (2006). Usage patterns of collaborative tagging systems. *Journal of Information Science, 32*(2), 198–208.

Guy, I., Zwerdling, N., Ronen, I., Carmel, D., & Uziel, E. (2010). Social media recommendation based on people and tags. In Proceedings of the 33rd international ACM SIGIR conference on research and development in information retrieval (pp. 194–201).

Han, J., Pei, J., & Yin, Y. (2004). Mining frequent patterns without candidate generation: A frequent-pattern tree approach. *Data Mining and Knowledge Discovery, 8*, 53–87.

Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems, 22*(1), 5–53.

Ji, A. -T, Yeon, C., Kim, H. -N., & Jo, G. -S. (2007). Collaborative tagging in recommender systems. In *Proceedings of the 20th Australian joint conference on artificial intelligence* (pp. 377–386).

Li, X., Guo, L., & Zhao, Y. (2008). Tag-based social interest discovery. In *Proceedings of the 17th international conference on World Wide Web* (pp. 675–684).

Liang, H., Xu, Y., Li, Y., Nayak, R., & Tao, X. (2010). Connecting users and items with weighted tags for personalized item recommendations. In *Proceedings of the 21st ACM conference on hypertext and hypermedia* (pp. 51–60).

Markines, B., Cattuto, C., Menczer, F., Benz, D., Hotho, A., & Stumme, G. (2009). Evaluating similarity measures for emergent semantics of social tagging. In *Proceedings of the 18th international conference on World Wide Web* (pp. 641–650).

McCallum, A., & Nigam, K. (1998). A comparison of event models for naïve Bayes text classification. In *Proceedings of AAAI-98 workshop on learning for text categorization* (pp. 41–48).

McLaughin, M. R., & Herlocker, J. L. (2004). A collaborative filtering algorithm and evaluation metric that accurately model the user experience. In *Proceedings of the 27th international ACM SIGIR conference on research and development in information retrieval* (pp. 329–336).

Milicevic, A. K., Nanopoulos, A., & Ivanovic, M. (2010). Social tagging in recommender systems: a survey of the state-of-the-art and possible extensions. *Artificial Intelligence Review, 33*(3), 187–209.

Nakamoto, R.Y., Nakajima, S., Miyazaki, J., Uemura, S., Kato, H., & Inagaki, Y. (2008). Reasonable tag-based collaborative filtering for social tagging systems. In *Proceedings of the 2nd ACM workshop on information credibility on the Web* (pp. 11–18).

Resnick, P., Iacovou, N., Suchak, M., Bergstorm, P., & Riedl, J. (1994). GroupLens: An open architecture for collaborative filtering of netnews. In *Proceedings of 1994 ACM conference on computer supported cooperative work* (pp. 175–186).

Sarwar, B., Karypis, G., Konstan, J., & Reidl, J. (2001). Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web* (pp. 285–295).

Schein, A. I., Popescul, A., Ungar, L. H., & Pennock, D. M. (2002). Methods and metrics for cold-start collaborative filtering. In *Proceedings of the 25th international ACM SIGIR conference on research and development in information retrieval* (pp. 253–260).

Sen, S., Vig, J., & Riedl, J. (2009). Tagommenders: Connecting users to items through tags. In *Proceedings of the 18th international conference on World Wide Web* (pp. 671–680).

Siersdorfer, S., & Sizov, S. (2009). Social recommender systems for web 2.0 folksonomies. In *Proceedings of the 20th ACM conference on hypertext and hypermedia* (pp. 261–270).

Tso-Sutter, K. H. L., Marinho, L. B., & Thieme, L. S. (2008). Tag-aware recommender systems by fusion of collaborative filtering algorithms. In *Proceedings of the 23rd ACM symposium on applied computing* (pp. 1995–1999).

Wang, J., Clements, M., Wang, J., De Vries, A. P., & Reinders, M. J. T. (2010). Personalization of tagging systems. *Information Processing and Management, 46*(1), 58–70.

Wetzker, R., Zimmermann, C., Bauckhage, C., & Albayrak, S. (2010). I Tag, You Tag: Translating tags for advanced user models. In *Proceedings of the 3rd ACM international conference on web search and data mining* (pp. 71–80).

Zhen, Y., Li, W. J., & Yeung, D. Y. (2009). TagiCoFi: Tag informed collaborative filtering. In *Proceedings of the 3rd ACM conference on recommender systems* (pp. 69–76).