Modelling User Behaviour and Interactions: Augmented Cognition on the Social Web

Ching-man Au Yeung and Tomoharu Iwata

NTT Communication Science Laboratories
2-4 Hikaridai, Seika-cho, Soraku-gun
Kyoto, 619-0237, Japan
{auyeung.chingman,iwata.tomoharu}@lab.ntt.co.jp

Abstract. Social sharing on the Web has become very popular in recent years. However, as the amount of information grows rapidly it becomes difficult for a user to discover relevant information. The principle of augmented cognition can be applied to help users on the Social Web. This can be done by modelling the behaviours and interactions of the users in a system in order to discover implicit relations among the users. We describe two related approaches to model user behaviours for different types of social sharing sites. We show that the methods can be used to help users identify social relations that are more important to them, as well as items that are more relevant to their interests.

1 Introduction

In recent years, Social Web applications have become very popular. Users establish social networks online and share their favourite items on the Web. However, as the amount of information and the size of one's social network grow rapidly, it becomes very difficult for a user to discover relevant information, or to know which acquaintance is more reliable as the source of relevant information. Augmented cognition aims at extending our ability to process information with computers. This principle can be applied to the Social Web by modelling the behaviours and interactions of the users in a system, discovering implicit relations among users and supporting users by recommendations.

We describe two related approaches to model user behaviours and interactions. Firstly, we consider systems where users may not have established any explicit social relation. We describe a probabilistic model [2] of how users choose to collect different items when influenced by different factors. Secondly, we consider product rating systems in which users have established explicit trust relations among themselves. We describe an extension to the standard matrix factorisation technique to estimate the strength of trust relations among users [1]. We also demonstrate that our methods give more accurate predictions of the actions and preferences of the users.

By analysing user behaviours and interactions, our methods can be used to reveal implicit social relations among users in Social Web applications. The results can help users identify social relations that are more important to them, and to retrieve information and items that are more relevant to their interests.

D.D. Schmorrow and C.M. Fidopiastis (Eds.): FAC 2011, HCII 2011, LNAI 6780, pp. 277–287, 2011.

In the next section, we briefly review works related to our research. In Section 3 and 4, we describe in detail our two proposed methods. We present some experimentation results in Section 5. Finally, we conclude the paper and mention some future research directions in Section 6.

2 Related Works

Online social networks have recently attracted the attention of researchers from different disciplines. While in the past it was difficult to collect data of social interactions among a large number of people, recently online social networking sites have provided rich data for studying real world social networks [4,12].

Several methods have been proposed to model social influences and information diffusion. In the common threshold model, a user would take an action if the number of his/her neighbours who have taken the same action reaches a certain threshold [5,7]. Song et al. [18,19], propose to use a Markov chain generated from the activity histories of the users to model information flow in a network. Tang et al. [20] introduce the notion of Topic Affinity Propagation to model social influence in a network with respect to different topics.

Our work is also related to research on recommendation systems based on social sharing data. For example, Shepitsen et al. [17] generate personalised recommendation by matching user profiles to clusters of tags obtained from a hierarchical clustering process. Bogers et al. [3] and Parra et al [16] present comparative studies of different collaborative filtering techniques on CiteULike. There are also attempts to make use of explicit social ties in social tagging systems to improve performance of collaborative filtering [8].

On the other hand, trust networks as a special form of social networks have also received much attention. They are usually implemented on product review or rating sites, on which making recommendations to users is an important application. For example, Ma et al. propose several different methods for incorporating trust relations in the standard matrix factorisation technique for collaborative filtering [14,13]. On the other hand, Jamali et al. [6] proposes TrustWalker, a random walk model that combines both trust-based recommendation and collaborative filtering based on item similarity.

Finally, there are a few works that focus on estimating the strength of trust and its effect on the opinions and ratings given by the users. For example, Matsuo and Yamamoto [15] present a hypothesis on the bi-directional effects between trust relations and item ratings. Focusing on more general social networks, Xiang et al. [22] proposes a generative model to estimate relationship strength in online social networks based on observed user interactions.

Overall, there is a substantial amount of works that investigate how user behaviours in social networks can be modelled. However, we believe that two issues have largely been overlooked. Firstly, when no explicit social ties are present, a method that can be used to model the implicit interactions among the users is not available. Secondly, when explicit relations have been established among the users, it is generally assumed that these relations truly reflect the similarity between the users. However, our study [1] shows that trust relations do not necessarily imply similarity. Hence a effective method for estimating the strength of trust relations is needed.

3 Implicit Influence in Social Sharing Sites

In this section, we focus on social sharing sites where users may not establish explicit ties among themselves. Examples of this kind include Delicious[1] and LibraryThing[2]. A social sharing system $\mathcal{S}$ of this kind can be defined as follows:

Definition 1. *A social sharing system $\mathcal{S}$ is a tuple $\mathcal{S} = \langle U, I, Y \rangle$. U is a set of users, I is a set of shared items among the users, and Y is a set of posts. A post $(u, i, t) \in Y$ represents the fact that $u \in U$ posts/adopts $i \in I$ at time t.*

To model user behaviours, we consider how users come to adopt certain items. On the one hand, users are free to introduce any new item to a social sharing site. On the other hand, users are also likely to discover something interesting in other users' collection. In many cases, social relations cannot be treated as the only means through with users influence one another. As a user's collection is publicly available in most cases, influence exists even when a social tie does not exist. In the following, we describe a model that explains how users adopt different items in a social sharing site under the influence of different factors.

3.1 A Model of Social Sharing

In our model, we assume that when a user decides to adopt an item by going through a two-step process: the user first selects a factor that would reveal to him/her a set of items, and then he/she chooses an item from that set. Here, the factor can be another user, the list of recent items or popular items, or even the user him/herself. Mathematically, the probability that user u would adopt item i at time t is defined as:

$$P(i|u,t) = \sum_{u' \in U} P(u'|u) G(i|u',t), \tag{1}$$

where $P(u'|u)$ represents the probability that u is influenced by the factor u' when he/she attempts to adopt something. $G(i|u', t)$ represents the probability that item i is chosen when factor u' is selected at time t.

The above definition is flexible and can be used to model a wide range of behaviours in different social sharing systems. When u' is a real user, $P(u'|u)$ represents the influence of u' on u. When u' is the list of popular items, $P(u'|u)$ represents how likely u would adopt something popular. Different factors can be modelled by defining a different $G(i|u', t)$. Below are some factors that are likely

[1] Delicious: `http://www.delicious.com/`

[2] LibraryThing: `http://www.librarything.com/`

to be found in a common social sharing system: (1) influence from other users, (2) list of recent/new items, (3) list of popular items, (4) random browsing of the items, and (5) 'Self-influence'.

For example, for the list of popular items, we can define $G(i|u', t)$ to be proportional to the number of users who have adopted i so far. $G(i|u', t)$ for other factors can be defined in a similar fashion. In addition, 'self-influence' refers to cases in which the user simply discovers an item external to the system and is the first one to adopt the item. In this case, we can assume that the user is influenced by him/herself. Let u_n be the factor of self-influence. In this case, instead of defining a particular $G(i|u_n, t)$, we can simply estimate $P(u_n|u)$ by the proportion of items of which u is the first user to adopt it in the system.

3.2 Parameter Estimation

The parameters of the model are the probabilities $P(u'|u)$. Given a history of user activities, we can estimate the parameters by maximising the log-likelihood of observed data under the constraint that $\sum_{u'} P(u'|u) = 1$. Note that if we model 'self-influence' as described above, we can estimate $P(u_n|u)$ in advance. Hence, our constraint becomes $\sum_{u'} P(u'|u) = 1 - P(u_n|u)$. Let U_A be the set of factors except u_n. The log-likelihood of the observed data is given by:

$$\log L = \sum_{(u,i,t)\in Y} \log \sum_{u'\in U_A} P(u'|u)G(i|u', t). \tag{2}$$

To estimate the parameters, we can employ the EM algorithm. In the E-step, we compute the posterior probability using the Bayes rule:

$$P(u'|u, i, t) = \frac{P(u'|u)G(i|u', t)}{\sum_{u''\in U_A} P(u''|u)G(i|u'', t)}. \tag{3}$$

In the M-step, we obtain the next estimate of the probabilities $P(u'|u)$ as follows:

$$P(u'|u) \propto \sum_{(u,i,t)\in Y} \sum_{i\in I} P(u'|u, i, t). \tag{4}$$

By iterating the above two steps until convergence, we obtain estimates for the probabilities $P(u'|u)$. As a result, we can generate recommendations to users by using $P(u'|u)$ (influential users) and $P(i|u, t)$ (items that the users may find interesting).

4 Strength of Trust Relations in Product Rating Sites

In this section, we turn our attention to social sharing sites in which users establish trust relations among themselves. In many proposals of using trust relations to generating recommendations, trust relations are usually taken at face value, i.e. it is usually assume that users who trust each other tend to have similar interests and opinions. However, our study [1] of a popular product review sites, Epinions, reveals that this is not usually the case. Hence, we believe it is necessary to estimate the true strength of the trust relations before utilising them in generating recommendations.

We propose an extension to matrix factorisation, which is commonly used to analyse user preferences in rating systems, to estimate the strength of trust relations among users. The basic matrix factorisation technique aims at revealing the latent factors that determine the ratings given by the users. Our extension allows us to explain user ratings by both latent factors as well as the different degree of influence they receive from their trusted neighbours. In other words, we consider matrix factorisation as a tool for *predicting ratings* as well as for *studying social relations among users*.

4.1 Estimating Trust by Matrix Factorisation

In general, a product rating system consists of a set U of users and a set I of items. Users express their interests or preferences in different items by rating the items with scores from a specific range. The interactions between the M users and the N items can be represented by an $M \times N$ matrix $\mathbf{R}$, where $M = |U|$ and $N = |I|$. An element r_{ui} in $\mathbf{R}$ indicates the rating of user u on item i. We represent the set of observed ratings as O.

Matrix factorisation aims at finding out the latent factors that can be used to explain the ratings given by the users. This is done by decomposing $\mathbf{R}$ into a $M \times K$ matrix $\mathbf{P}$ and a $N \times K$ matrix $\mathbf{Q}$, where K is the number of latent factors. Here, we extend standard matrix factorisation by considering trust relations among the users.

Let $\mathbf{G}$ be an $M \times M$ matrix encoding the trust network that is established *explicitly* by the users themselves. g_{uv}, an element in $\mathbf{G}$, equals to 1 if u trusts v, and 0 otherwise. In addition, we let $\mathbf{S}$ be an $M \times M$ matrix that holds the *estimated* trust relations: $s_{uv} = 0$ if $g_{uv} = 0$, and $s_{uv} \geq 0$ if $n_{uv} = 1$. The values of s_{uv} will be estimated in the matrix factorisation process, and they represent the strengths of the trust relations among the users.

Thus, for a particular tuple (u, i, r), two factors are at play in determining the rating r. Firstly, the rating is determined by the latent factor model. Secondly, u gives i a particular rating because he/she is influenced by some users he/she trusts. If the users trusted by u give high ratings, u should also tend to give high ratings.[3] Based on the above idea, a rating r_{ui} in the $\mathbf{R}$ can be estimated by:

$$\widehat{r}_{ui} = \alpha \sum_{k=1}^{K} p_{uk} q_{ik} + (1 - \alpha) \frac{\sum_{\forall v, g_{uv} > 0} s_{uv} r_{vi}}{\sum_{\forall v, g_{uv} > 0} s_{uv}} \tag{5}$$

where p_{uk} and q_{ik} are elements of the matrices $\mathbf{P}$ and $\mathbf{Q}$ respectively, and α is a parameter than controls the contributions of the two factors. In this model, the

[3] When a user trusts no other users, we assume that he/she trusts a virtual user who has rated all items by their respective mean ratings.

values of s_{uv} are to be estimated based on the differences between ratings given by pairs of users. If two users give very different ratings to the same products, s_{uv} will be small even if a trust relation exists between them.

4.2 Parameter Estimation

In this model, the parameters are p_{uk}, q_{ik}, and s_{uv}. To estimate the values of these parameters, we solve an optimisation problem that involves minimising the following regularised sum-of-squared error:

$$\min \quad \frac{1}{2} \sum_{(u,i,r)\in O} (r_{ui} - \alpha \sum_{k=1}^{K} p_{uk} q_{ik} - (1-\alpha) \frac{\sum_{\forall v, g_{uv}>0} s_{uv} r_{ui}}{\sum_{\forall v, g_{uv}>0} s_{uv}})^2 + \frac{\beta}{2}(\sum_{u,k} p_{uk}^2 + \sum_{i,k} q_{ik}^2), \tag{6}$$

where the last component is a regularisation term to avoid the parameters from taking on large values that might result in overfitting.

In our implementation, we use gradient descent to solve this optimisation problem. Parameters are initialised with random values. Let $e_{ui} = r_{ui} - \widehat{r}_{ui}$ be the error of estimation, γ be a learning meta-parameter, and there be a constraint that $\sum_v s_{uv} = 1$, the followings are the update rules for the different parameters:

$$p_{uk} \leftarrow p_{uk} + \gamma(\alpha \cdot e_{ui} \cdot q_{ik} - \beta \cdot p_{uk}) \tag{7}$$

$$q_{ik} \leftarrow q_{ik} + \gamma(\alpha \cdot e_{ui} \cdot p_{uk} - \beta \cdot q_{ik}) \tag{8}$$

$$s_{uv} \leftarrow s_{uv} + \gamma((1-\alpha) \cdot e_{ui} \cdot \frac{r_{vi} \sum_{\forall v, g_{uv}>0} s_{uv} - \sum_{\forall v, g_{uv}>0} s_{uv} r_{vi}}{(\sum_{\forall v, g_{uv}>0} s_{uv})^2}). \tag{9}$$

At the end of the training period, we should obtain a weight s_{uv} for each trust relations established between some users u and v. These can be considered as the true strengths of trust relations among the users. Together with the learnt values of the matrices $\mathbf{P}$ and $\mathbf{Q}$, we can also generate predictions for unknown ratings.

5 Experiments

To study the effectiveness of our proposed methods, we carry out experiments by using datasets collected from two popular social sharing sites, namely Delicious and Epinions.

5.1 Capturing Implicit Influence in Delicious

Delicious is a popular social bookmarking site. We use a dataset that is publicly available for research purpose [21].[4] It contains bookmarking histories of

[4] `http://www.dai-labor.de/index.php?id=1726`

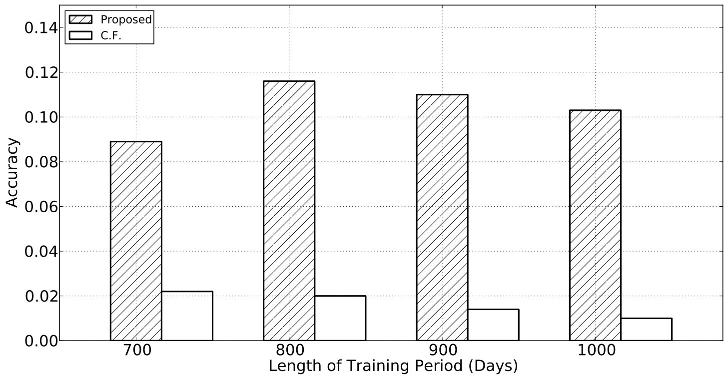

Fig. 1. Average accuracy of the model under different conditions across all 50 tags, compared to simple collaborative filtering

over 950,000 users and about 50 million bookmarks, spanning the period from September 2003 to December 2007. To avoid data sparsity, we remove users who have only adopted one item. In addition, we choose to train our model on the datasets of the 50 most frequently used tags in the dataset.

In this experiment, we study whether the probabilities $P(u'|u)$ can be used to generate accurate recommendations to the users. Firstly, we split the dataset into two parts, respecting the chronological order of the observations. We train the model using adoption histories of users in the first part, and test our model on those in the second part. For a particular user u, using the probabilities $P(u'|u)$ obtained from the training process, we predict the next items that will be adopted by u after time t.

Recommendation is done by first calculating the probability $P(d|u,t)$ using Equation 1 and then ranking the items in descending order of their probability of being adopted by the user u. We consider it successful if the next items adopted by the user appear in the top m results of the ranking. We define *accuracy*, our performance measure, as the average proportion of items that are adopted by the user and at the same time appear in our top results.

The model under evaluation incorporates all the five factors mentioned in Section 3.1. We train a separate model for each of the 50 datasets. We test our model by using different amount of training data (i.e. data from the first 700, 800, 900 and 1,000 days). For each dataset, we randomly sample 1,000 users and collect the next 10 items they adopt in the testing period. Users and items that did not appear in the training period are ignored. We then use the probabilities $P(u'|u)$ obtained in the training period and Equation 1 to come up with a ranked list of items. In our experiment we set $m = 50$. For the purpose of comparison, we implement a simple k-nearest-neighbour collaborative filtering (C.F.) method.

Figure 1 show the result of our experiment. We observe our model can be used to predict item adoption at a much higher accuracy when compared with the simple collaborative filtering algorithm. In other words, for a particular user, other users who have similar adoption histories do not necessarily possess items that are interesting to him/her. Instead, users who are found to be influential to a particular user are useful for predicting item adoption.

Our results reveal an interesting fact about the importance of considering the temporal order of adoption in making recommendations. Collaborative filtering does not consider this order and therefore is not able to distinguish between *followers* and *influencers*. However, this distinction is important because followers are more likely to adopt items from influencers but not vice versa. The probabilities $P(u'|u)$, which is asymmetric for a pair of users, is able to model this distinction, and as a result is able to generate more accurate recommendations.

5.2 Estimating Strength of Trust Relations in Epinions

Next, we conduct an experiment using data collected from Epinions, in order to evaluate our proposed matrix factorisation method. Epinions is a popular product rating sites. Users write reviews and give ratings to a wide rane of products. From the Web site of Epinions, we collect over 900,000 ratings given by about 60,000 users. We also collect all the existing trust relations among these users.

Regarding the experiment, it should be noted that there are many matrix factorisation techniques that are shown to achieve very high accuracy in predicting ratings. Our objective is not to compete with the state-of-the-art algorithms. Instead, we want to demonstrate that our method can be used estimate strength of trust relations among users, which helps us to generate more accurate rating predictions.

For comparison, we consider predictions made by the standard matrix factorisation method, which only considers the latent factors but not the social relations among the users. The metric used to measure performance is the standard root-mean-squared-error (RMSE):

$$RMSE = \sqrt{\frac{\sum_{(u,i,r)\in T}(r_{ui} - \widehat{r}_{ui})^2}{|T|}} \tag{10}$$

where T is the set of testing data (a set of ratings). A lower RMSE means that the predictions are more accurate.

An important parameter in our model is α, which controls the ratio of contributions from the latent factor model and the trust relations. We test different value of α to investigate its effect on performance. We set the other parameters as follows: $\gamma = 0.001$ (learning meta-parameter), $\beta = 0.01$ (regularisation) and $K = 20$ (number of latent factors). Results of using 80% of the datasets as training data and the rest as testing data are presented in Figure 2.

We can observe that our proposed method constantly achieve lower RMSE than the standard matrix factorisation method. This shows that the estimated strengths of trust relations contribute to more accurate predictions. Our proposed method performs better when α is larger (in the range of [0.5,0.8]). A larger α means that a higher weight would be put on the latent factors.

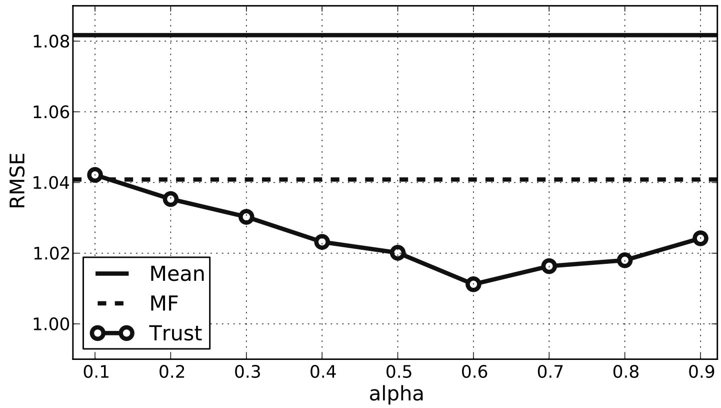

Fig. 2. RMSE of rating predictions in Epinions. 'Mean' refers to predicting a rating using the mean of existing ratings of an item. 'MF' refers to the standard matrix factorisation method. 'Trust' refers to our proposed method.

Therefore, it suggests that the latent factors are still important in generating accurate predictions. However, it also shows that an appropriate combination of the two components is crucial to achieving higher accuracy.

The optimal value of α probably depends on the specific characteristics of the system under study. In addition of setting α manually, it is possible to estimate α in the training process, or even assign different α values to different users. We plan to investigate these extensions in the future.

6 Conclusions

In this paper, we presented two methods for modelling user behaviours in social sharing sites. The first one can be used to model probabilistically user behaviours in social sharing sites in which explicit social ties are not available. It can be used to discover influential users and generate recommendations. The second one targets review and rating sites in which users maintain a trust network among themselves. The method based on matrix factorisation can be used to estimate the true strength of their trust relations by analysing their common ratings. It can also be used to predict product ratings given by the users more accurately, compared to methods that do not consider the strength of trust relations.

Our objective in this paper is to apply the principle of augmented cognition to assist users in processing huge amount of information and social ties in online social networking systems. In many cases, users can only maintain a flat list of social relations, and cannot really distinguish between acquaintances who share information that is more relevant to themselves and those who are less likely to do so. The methods presented in this paper thus augment the ability of the users to handle online relations and to discover information that is more relevant to their interests.

There are several directions for extending this work. The first direction, common to both methods, is to consider variations of the strength of (implicit) relations across different topics and contexts. For example, user A may be influenced by user B in terms of electronic goods, but by user C instead in terms of movies. In other words, there is a need to consider different contexts when modelling user behaviours. In addition, in both cases it would be interesting to rank the users by how influential they are. Hence, we are planning to develop user ranking algorithms based on the results produced by these two methods.

References

1. Au Yeung, C., Iwata, T.: Strength of social influence in trust networks in product review sites. In: WSDM 2011, ACM Press, New York (2011)
2. Au Yeung, C., Iwata, T.: Capturing Implicit Influence in Online Social Sharing. In: HT 2010, ACM Press, New York (2010)
3. Bogers, T., van den Bosch, A.: Recommending scientific articles using citeulike. In: RecSys 2008, pp. 287–290. ACM Press, New York (2008)
4. Crandall, D., Cosley, D., Huttenlocher, D., Kleinberg, J., Suri, S.: Feedback effects between similarity and social influence in online communities. In: KDD 2008, pp. 160–168. ACM Press, New York (2008)
5. Goyal, A., Bonchi, F., Lakshmanan, L.V.S.: Learning influence probabilities in social networks. In: WSDM 2010, ACM Press, New York (2010)
6. Jamali, M., Ester, M.: Trustwalker: a random walk model for combining trust-based and item-based recommendation. In: KDD 2009, pp. 397–406. ACM Press, New York (2009)
7. Kempe, D., Kleinberg, J., Tardos, E.: Maximizing the spread of influence through a social network. In: KDD 2003, pp. 137–146. ACM Press, New York (2003)
8. Konstas, I., Stathopoulos, V., Jose, J.M.: On social networks and collaborative recommendation. In: SIGIR 2009, pp. 195–202. ACM Press, New York (2009)
9. Koren, Y.: Factorization meets the neighborhood: a multifaceted collaborative filtering model. In: KDD 2008, pp. 426–434. ACM Press, New York (2008)
10. Koren, Y.: Collaborative filtering with temporal dynamics. In: KDD 2009, pp. 447–456. ACM Press, New York (2009)
11. Leskovec, J., Adamic, L.A., Huberman, B.A.: The dynamics of viral marketing. ACM Trans. Web 1(1), 5 (2007)
12. Leskovec, J., Singh, A., Kleinberg, J.M.: Patterns of influence in a recommendation network. In: Ng, W.-K., Kitsuregawa, M., Li, J., Chang, K. (eds.) PAKDD 2006. LNCS (LNAI), vol. 3918, pp. 380–389. Springer, Heidelberg (2006)
13. Ma, H., Lyu, M.R., King, I.: Learning to recommend with trust and distrust relationships. In: RecSys 2009, pp. 189–196. ACM Press, New York (2009)
14. Ma, H., Yang, H., Lyu, M.R., King, I.: Sorec: social recommendation using probabilistic matrix factorization. In: CIKM 2008, pp. 931–940. ACM Press, New York (2008)
15. Matsuo, Y., Yamamoto, H.: Community gravity: measuring bidirectional effects by trust and rating on online social networks. In: WWW 2009, pp. 751–760. ACM Press, New York (2009)
16. Parra, D., Brusilovsky, P.: Collaborative filtering for social tagging systems: an experiment with citeulike. In: RecSys 2009, pp. 237–240. ACM Press, New York (2009)

17. Shepitsen, A., Gemmell, J., Mobasher, B., Burke, R.: Personalized recommendation in social tagging systems using hierarchical clustering. In: RecSys 2008, pp. 259–266. ACM Press, New York (2008)
18. Song, X., Tseng, B.L., Lin, C.-Y., Sun, M.-T.: Personalized recommendation driven by information flow. In: SIGIR 2006, pp. 509–516. ACM Press, New York (2006)
19. Song, X., Chi, Y., Hino, K., Tseng, B.L.: Information flow modeling based on diffusion rate for prediction and ranking. In: WWW 2007, pp. 191–200. ACM Press, New York (2007)
20. Tang, J., Sun, J., Wang, C., Yang, Z.: Social influence analysis in large-scale networks. In: KDD 2009, pp. 807–816. ACM Press, New York (2009)
21. Wetzker, R., Zimmermann, C., Bauckhage, C.: Analyzing social bookmarking systems: A del.icio.us cookbook. In: Proc. of Mining Social Data Workshop, collocated with ECAI 2008, pp. 26–30 (2008)
22. Xiang, R., Neville, J., Rogati, M.: Modeling relationship strength in online social networks. In: WWW 2010, pp. 981–990. ACM Press, New York (2010)