An Adaptive Method for the Tag-Rating-Based Recommender System

Xi Yuan and Jia-jin Huang

International WIC Institute, Beijing University of Technology, China
yuanxi.gloria@yahoo.com.cn, hjj@emails.bjut.edu.cn

Abstract. In this paper, we propose an adaptive method for recommender system based on users' preference to items represented by the ratings of users. This method defines a term-association matrix to describe the relation between tags and items properties. A gradient descent method is employed to compute the association matrix. The association matrix is then used to implement the two kinds of recommendation, namely, tag recommendation and items properties recommendation.

1 Introduction

In Web 2.0, the overload information makes Web users difficult to find useful Web information. Recommender systems provide available methods to help users be free from large data by predicting and recommending information in which Web users may be interested.

Collaborative filtering (CF) recommendation is one of the most successful techniques. CF recommends items among people with similar tastes [2,3]. Besides, content-based recommendation (BCR) can provide recommendations by encoding users' preferences from textual information [6]. Hybrid recommendation [8,12] combines CF and BCR into a single integrated model [4].

The advent of Web 2.0 brings a new form of user-centric method, folksonomy [10], which allows users not only to tag items for their own characters with user-defined words but also to upload items to express their opinions. The tag-based recommender system has attracted more attention recently.

How to create the Web user interest model is the key issue of the tag-based recommender systems. The main directions for the research can be divided into matrix-based methods, clustering-based methods and graph-based methods. In matrix-based methods, Xu et al [14] proposed latent semantic analysis (LSA) to compute the included-angle cosine between tags and items by using tag-item matrix. Xu et al [15] further introduced higher-order singular value decomposition (HOSVD) to improve recommendation quality and stability, which finds association between users, items and tags by combining them into a framework. With respect to clustering-based methods, Reyn et al [9] constructed a scenario-based CF model based on the similarity of tags, which recommends items by abstracting tags to the user vectors and counting users similarity. Jonathan et al [5] applied the TF-IDF formula to cluster tags by a hierarchical method, and

R. Huang et al. (Eds.): AMT 2012, LNCS 7669, pp. 206–214, 2012.

constructed the Web user interest model by users' interests on items. As for graph-based methods, Andreas et al [1] proposed a modified PageRank algorithm, namely FolkRank, which consider the connection between users, tags and items to an undirected graph. Guan et al [17] proposed a framework based on graph Laplacian to model interrelated multi-type objects.

Although the social tag is especially useful for both searching and organizing items, many studies argue that not all the tags benefit recommendation [16] because the unrestricted nature of the tagging function is liberating [7]. In tagging systems, tags with free style will interfere the analysis of the structure and users' behaviors.

Currently, ratings have been regarded as an effective and simple form of recommendation. Tags represent users' interests and preferences in more detail, but the tags with free style also interfere the analysis of user behaviors and system structure. For these reasons, we construct the Web interest model by combining tags with ratings, and construct a tag-rating-based term-association matrix for high-efficiency recommendation.

2 The Web User Interest Model of the Tag-Rating-Based Recommender System

2.1 A User-Item-Tag-Rating Fourfold Graph and the Vector Space Model

In the tag-rating-based recommender system, the recommender system consists of users, items, tags and ratings [18]. Users express their preferences by tagging items with high ratings.

Let $D = (U, I, T, R)$ denote the four parts. The component $U = \{u_1, u_2, ..., u_k\}$ is the set of users, each user $u_i (1 \leq i \leq k)$ is modeled as a vector $\mathbf{u_i} = (\mu_1, \mu_2, ...\mu_m)$ over the set of tags $T = \{t_1, t_2, ..., t_m\}$, and each tag $t_s (1 \leq s \leq m)$ is represented by the weight $\mu_s (1 \leq s \leq m)$ of $\mathbf{u_i}$; $I = \{i_1, i_2, ..., i_p\}$ is the set of items, each item $i_j (1 \leq j \leq p)$ is modeled as a vector $i_j = (\nu_1, \nu_2, ..., \nu_n)$ over the set of item properties $G = \{g_1, g_2, ..., g_n\}$, and each property $g_t (1 \leq t \leq n)$ is represented by the weight $\nu_t (1 \leq t \leq n)$ of $\mathbf{i_j}$; $R = \{r_{11}, r_{12}, .., r_{ij}, ..., r_{kp}\}$ is the set of ratings, each r_{ij} presented the rating of item i_j by user u_i. An example is shown in Fig. 1.

In this paper, the rating relation is used to describe users' preference to items. Each rating relation is measured by ratings given by users. In the next section, we will describe the definition of rating relation and use it to construct the Web user interest model.

2.2 The Definition of Rating Relation

In this section, we will introduce the definition of rating relation in two kinds of recommendation which can be used to measure user preference to items in the tag-rating-based recommender system.

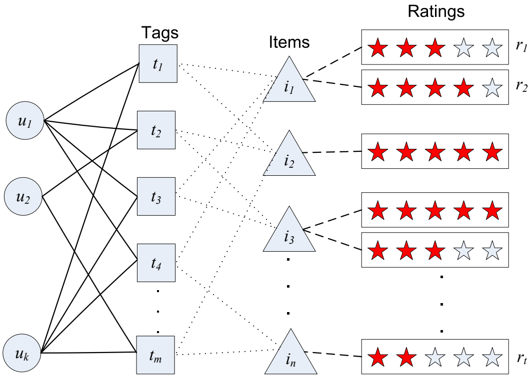

Fig. 1. The illustration of a user-tag-item-rating fourfold graph

The (u, i, t, r) is defined to a user-item-tag-rating $(UITR)$ quaternion. Given a tag, for $i, i' \in I$ and $u, u' \in U$. The rating relation is defined as follows:

$$(u, i, t) \succ (u', i', t') \Leftrightarrow \textit{The rating of item } i \textit{ with } t \textit{ by user } u \textit{ is higher than } i' \textit{ with } t' \textit{ by user } u'. \tag{1}$$

In the tag-rating-based recommender system model, the components rely on 2-dimensional projections of the user-tag-item (UTI) matrix, which reduce the dimensionality of the data but sacrifice its informational content. We produce an item-user (IU) matrix projection and a user-item (UI) matrix projection. We define them to be binary, which the users and the items vectors are indicated by whether or not the user has annotated the tag and the item has classified to the property. The weights $\mu_s (1 \le s \le m)$ and $\nu_t (1 \le t \le n)$ have only two values, 1 or 0. Thus, we propose two kinds of recommendation: UI recommendation and IU recommendation which recommend items properties and tags respectively.

So, we separate the Eq. (1) for two kinds of recommendation, and define the individual rating relation as follows:

For UI recommendation, the rating by users on particular item and tag can be specified by individual rating relation as follows:

$$\begin{aligned} u \succ_{i,t} u' &\Leftrightarrow \textit{The rating of item } i \textit{ with tag } t \textit{ by user } u \textit{ is higher than } u'. \\ &\Leftrightarrow (u, i, t) \succ (u', i, t) \end{aligned} \tag{2}$$

For IU recommendation, the rating of items on particular user and tag can be specified by individual rating relation as follows:

$$\begin{aligned} i \succ_{i,t} i' &\Leftrightarrow \textit{The rating of item } i \textit{ with tag } t \textit{ by user } u \textit{ is higher than } i'. \\ &\Leftrightarrow (u, i, t) \succ (u, i', t) \end{aligned} \tag{3}$$

The subscript of Eq. (2) and Eq. (3) emphasized the fact that the rating relation is defined to particular item, user and tag.

2.3 The Web User Interest Model of the Tag-Rating-Based Recommendation System

In this section, based on the algorithm proposed by Wong et al [13] combined with the social tag, we propose a Web interest model by constructing IU term-association matrix and UI term-association matrix. In order to be short, we only introduce the construction of IU term-association matrix, the construction of UI term-association matrix is similar.

We construct an IU term-association matrix to describe the relation between users and items by a bilinear function:

$$g(i,u) = \sum_{i=1}^{n}\sum_{j=1}^{m} \nu_i a_{ij} \mu_j = \mathbf{iAu^T} \tag{4}$$

where a_{ij} measures the strength of association between item property g_i and tag t_j, and $\mathbf{A} = (a_{ij})$ is the IU term-association matrix, which is not necessarily a symmetric matrix, rows and columns of $\mathbf{A}$ is determined by the dimension of the item properties and the tags. The construct matrix $\mathbf{A}$ need to satisfy the condition: for any $UITR$ quaternion $(u,i,t,r),(u,i',t,r') \in UITR$,

$$i \succ_{u,t} i' \Rightarrow \mathbf{iAu^T} >_{\mathbf{t}} \mathbf{i'Au^T} \tag{5}$$

We called $\mathbf{w} = \mathbf{i} - \mathbf{i'}$ is a difference vector, so the condition $\mathbf{iAu^T} >_{\mathbf{t}} \mathbf{i'Au^T}$ can be written as $\mathbf{wAu^T}$, the set W consisting of item rating vector teams defined by:

$$W = \{(\mathbf{w},\mathbf{u}) | \mathbf{w} = \mathbf{i} - \mathbf{i'}, i \succ_{u,t} i'\} \tag{6}$$

Obviously, the problem of finding the IU term-association matrix $\mathbf{A}$ satisfying Eq. (5) is reduced to a problem of finding a solution matrix to satisfying condition as follows:

$$\mathbf{wAu^T} >_t 0, (\mathbf{w},\mathbf{u}) \in W \tag{7}$$

Borrowing the algorithm in [13], we will introduce the procedure of calculating the IU term-association matrix.

(i) We start with an initial matrix $\mathbf{A^{(0)}}$ and let $k = 0$, usually matrix $\mathbf{A^{(0)}}$ is an unit matrix.

(ii) If $\mathbf{wAu^T} >_t 0$, we say that matrix $\mathbf{A}$ correctly describes the rating relationship $i \succ_{u,t} i'$, so we defined:

$$\Gamma(\mathbf{A^{(k)}}) = \{(\mathbf{w},\mathbf{u}) | (\mathbf{w},\mathbf{u}) \in W \wedge \mathbf{wA^{(k)}u^T} \le 0\} \subseteq W \tag{8}$$

If $\Gamma(\mathbf{A^{(k)}}) = \Phi$, terminate the procedure.

(iii) $\Gamma(\mathbf{A^{(k)}})$ is the term-association matrix in the $(k+1)th$ iteration, which is obtained by the gradient descent method, and calculated by:

$$\mathbf{A^{(k+1)}} = \mathbf{A^{(k)}} + [\sum_{(\mathbf{w},\mathbf{u})\in\boldsymbol{\Gamma}(\mathbf{A^{(k)}})} \mathbf{w}]^{\mathbf{T}}\mathbf{u} \tag{9}$$

(iv) Let $k = k + 1$, go back to (ii).

If the solution matrix exists, we can find it by the algorithm above. But in practice it is difficult to find the solution matrix $\mathbf{A}$ satisfying the termination condition (ii), because the increased complexity of the data increase the complexity of the matrix. Thus, we find the solution matrix $\mathbf{A}$ by calculating the accuracy of the recommendation when $k = 0, 1, 2, 3$ and the term-association matrix is $\mathbf{A^1}, \mathbf{A^2}, \mathbf{A^3}$ respectively.

This algorithm provides a systematic method to construct term-association, and needn't to introduction any particular parameters. For the UI recommendation, all we need is to exchange the place of $\mathbf{w}$ and $\mathbf{u}$ and use $\mathbf{n} = \mathbf{u} - \mathbf{u}'$ instead of $\mathbf{w} = \mathbf{i} - \mathbf{i}'$. Tags can be recommended by the IU algorithm above in the IU recommendation and for the UI recommendation, item properties could be recommended by the UI term-association matrix.

2.4 The Network Configuration of the Tag-Rating-Based Recommender System

By constructing the Web interest model above, tags can be recommended by finding IU term-association matrix $\mathbf{A}$. In this section, we describe the network configuration of the tag-rating-based recommender system and give the algorithm of calculating the IU term-association matrix $\mathbf{A}$.

Figure 2 shows the network configuration. Two directions of the network represent the processing of the UI recommendation and the IU recommendation, which term-association matrixes calculated by $g(u, i) = \mathbf{uAi^T}$ and $g(i, u) = \mathbf{iAu^T}$, respectively.

For the IU recommendation, the input layer is represented by an item node i. The node i connects property node g_i by an individual weight μ_i . The output layer consists of a node which pools the user terms with individual weight $\beta_j = 1$. For the UI recommendation, the input layer is represented by a user node u, which connects tag node t_j by an individual weight. The output layer consists of a node which pools the item terms with individual weight $\alpha_i = 1$. The weight between item property g_i and tag t_j is represented by a_{ij} and a_{ji}, which is the element of the matrix $\mathbf{A}$.

3 Experimental Evaluation

In this section we describe the methods used to gather and pre-process our datasets and our evaluation metrics. The algorithm above is used as two kinds of recommendation. This section mainly present the recommendation results.

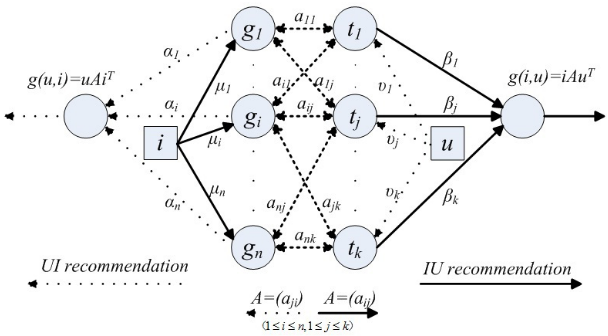

Fig. 2. The tag-rating-based recommender system network configuration

3.1 Dataset and Data Pre-processing

Our experiments are conducted by using the $MovieLens$ ($Version1.0$ ($May2011$)) dataset which are gathered from the corresponding $MovieLens$ Web site. Users are allowed to rating and tag movies on $MovieLens$ Web site. The dataset contains users, movies, tags and ratings, which need to remove rarely-occurring information to reduce the noise in the data.

The data of users, movies, tags, ratings are gathered into one data which the movies are more than 20 for each user, and are expressed by (u, i, t, r) quaternion. The data is resulted in 29694 annotations with 185 users, 4238 movies and 6570 tags, and each movie can be classified into several genres from 20. The set of tags $T = \{t_1, t_2, ..., t_m\}$ and the set of item properties $G = \{g_1, g_2, ..., g_n\}$ are respectively represented by the tags and the movie genres in the dataset, which weight are indicated by whether or not the movie has classified to the genre and the user has annotated the tag. Most important, the rating vector teams of IU and UI recommendation are need to be extracted from the (u, i, t, r) quaternion to Adata dataset and Bdata dataset. If $r > r'$, the rating relations are $i \succ_{u,t} i'$ and $u \succ_{i,t} u'$.

In the experiment, the training and test sets are created by 10-fold cross-validation method [11]. Firstly, each dataset is randomly split into 10 mutually exclusive subsets of approximately equal size. Each time the test set T_k is clustered by two randomly subsets and the corresponding training set TR_k is clustered by the other eight subsets, 10 times $(1 \le k \le 10)$ in all. Finally, we get 10 test sets $(T_1, T_2, ..., T_{10})$ and 10 training set $(TR_1, TR_2, ..., TR_{10})$ for each dataset.

For IU recommendation, the top 20% tags that calculated by $\mathbf{i} \times \mathbf{A} = \mathbf{u}$ are chosen to be the $finalTags$, and the tags that tagged for m_i in the training set are chosen to be the $targetTags$. The $finalTags$ and the $targetTags$ have eliminated duplicate entries. The accuracy of i_i for each time is gotten by: for $1 \le k \le 10$,

$$Acc_k(i_i) = \frac{NUM(finalTags \cap targetTags)}{NUM(targetTags)} \tag{10}$$

Table 1. The accuracy of the *IU* recommendation

items	i_{25}	i_{785}	i_{1219}	i_{2076}	i_{2730}	i_{3676}	i_{4194}	i_{5390}	i_{6258}	i_{6669}
$k=1$	57.38	59.26	51.55	66.67	57.81	60.14	52.78	57.78	53.84	57.41
$k=2$	60.00	63.09	53.13	70.00	60.95	63.54	54.73	60.29	57.16	60.00
$k=3$	65.10	66.71	60.00	70.83	66.16	67.27	61.80	65.97	63.09	65.87
$k=4$	68.49	69.20	63.37	72.39	68.65	69.63	64.60	68.57	65.85	68.49
$k=5$	68.89	69.75	63.90	73.57	69.10	69.96	65.68	69.07	66.46	69.02
$k=6$	62.38	63.05	57.38	68.78	62.67	63.09	58.33	62.50	59.73	62.47

The average accuracy of i_i is gotten by Eq. (11):

$$Ave(i_i) = \frac{\sum_{k=1}^{N} Acc_k(i_i)}{N}, (1 \leq N \leq 10) \tag{11}$$

For UI recommendation, the top 20% genres that calculated by $\mathbf{u} \times \mathbf{A} = \mathbf{i}$ are chosen to be the $finalGenres$, and the genres that classified to the movies tagged by u_j are chosen to be the $targetGenres$. Noticeable, the $finalGenres$ and the $targetGenres$ have eliminated duplicate entries. The accuracy of u_j for each time is gotten by: for $1 \leq k \leq 10$,

$$Acc_k(u_j) = \frac{NUM(finalGenres \cap targetGenres)}{NUM(targetGenres)} \tag{12}$$

The average accuracy of u_j is gotten by Eq. (13):

$$Ave(u_j) = \frac{\sum_{k=1}^{N} Acc_k(u_j)}{N}, (1 \leq N \leq 10) \tag{13}$$

3.2 Results and Discussion

The IU recommendation recommend tags for each movie by IU term-association matrix $\mathbf{A}$, which is calculated by Eq. (8). However, in practice, it is difficult to find the solution matrix, so we let $k = 1, 2, ...$ find the result matrix $\mathbf{A}$ with best recommendation effect. The average accuracies of selected movie are shown in Table 1. It's easy to find that the curve of $k = 5$ has the best accuracy for

Table 2. The accuracy of the *UI* recommendation

users	u_{2643}	u_{8787}	u_{12265}	u_{19923}	u_{23172}	u_{38662}	u_{45290}	u_{51954}	u_{68228}	u_{71331}
$k=1$	40.40	39.48	28.92	14.95	39.51	16.67	36.88	25.16	22.86	18.43
$k=2$	50.00	41.67	32.20	17.50	41.67	20.00	40.40	28.72	26.45	20.98
$k=3$	77.07	65.70	47.79	30.48	66.02	32.22	59.85	42.78	40.40	34.32
$k=4$	79.54	71.56	56.89	42.42	74.07	46.10	69.06	53.11	50.95	47.41
$k=5$	78.53	68.57	54.04	41.67	71.90	43.40	66.67	51.58	50.00	45.05

average 60%, which means that the recommendation is the best when $\mathbf{A^5}$ is the result matrix.

In this paper, tags are freely assigned by users, which results in much noise of tags. The next task is using the clustering methods and anti-spam technique to reduce the noise and provide high-quality recommendations.

Similar to the *IU* recommendation, finding the result matrix by setting $k = 1, 2, ...,$ the *UI* recommendation gives a best average accuracy of 50% when $k = 4$. The average accuracies of selected user are shown in Table 2.

4 Conclusion

In this paper, we have constructed the Web user interest model and the network configuration of the tag-rating-based recommendation system by combining the rating relation with user-item-tag vector teams. We provide two kinds of recommendation strategies, namely the *IU* recommendation and the *UI* recommendation, which can be used to provide tags recommendation and item properties recommendation respectively. The relation between items and users is represented by a term-association matrix **A**. Given items, the *IU* recommendation calculates weighs of tags and selects the top 20% tags to recommend. Given users, the *UI* recommendation calculates weighs of item properties and selects the top 20% item properties to recommend. The results show that the algorithms are effective. The next task is constructing multi-layer Web interest model for diversity recommendation and to improve the recommendation qualities.

Acknowledgements. This work is supported by Beijing Natural Science Foundation (4102007), the CAS/SAFEA International Partnership Program for Creative Research Teams, the China Postdoctoral Science Foundation Funded Project (2012M510298), Projected by Beijing Postdoctoral Research Foundation (2012ZZ-04), and the doctor foundation of Beijing University of Technology (X0002020201101).

References

1. Hotho, A., Jäschke, R., Schmitz, C., Stumme, G.: Information Retrieval in Folksonomies: Search and Ranking. In: Sure, Y., Domingue, J. (eds.) ESWC 2006. LNCS, vol. 4011, pp. 411–426. Springer, Heidelberg (2006)
2. David, G., David, N., Brain, M.O., Douglas, T.: Using Collaborative Filtering to Weave an Information Tapestry. Communication of the ACM-Special Issue on Information Filtering 35(12), 61–70 (1992)
3. John, S.B., David, H., Carl, K.: Empirical Analysis of Predictive Algorithms for Collaborative Filtering. In: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence (UAI 1998), pp. 43–52 (1998)
4. Jonathan, G., Thomas, S., Bamshad, M.: Resource Recommendation in Social Annotation Systems: A Linear-Weighted Hybrid Approach. Journal of Computer and System Sciences 78(4), 1160–1174 (2012)

5. Gemmell, J., Shepitsen, A., Mobasher, B., Burke, R.: Personalizing Navigation in Folksonomies Using Hierarchical Tag Clustering. In: Song, I.-Y., Eder, J., Nguyen, T.M. (eds.) DaWaK 2008. LNCS, vol. 5182, pp. 196–205. Springer, Heidelberg (2008)
6. Marko, B., Yoav, S.: Content-Based Collaborative Recommendation. Communications of the ACM 40(3), 66–72 (1997)
7. Mathes, A.: Folksonomies-Cooperative Classification and Communication Through Shared Metadata. Computer Mediated Communication (2004)
8. Qing, L.: Clustering Approach for Hybrid Recommender System. In: Proceedings of the 2003 IEEE/WIC International Conference on Web Intelligence (WI 2003), pp. 33–38 (2003)
9. Reyn, N., Shinsuke, N., Jun, M., Shunsuke, U.: Tag-based Contextual Collaborative Filtering. IAENG International Journal of Computer Science 34(2), 214–219 (2007)
10. Jäschke, R., Marinho, L., Hotho, A., Schmidt-Thieme, L., Stumme, G.: Tag Recommendations in Folksonomies. In: Kok, J.N., Koronacki, J., Lopez de Mantaras, R., Matwin, S., Mladenič, D., Skowron, A. (eds.) PKDD 2007. LNCS (LNAI), vol. 4702, pp. 506–514. Springer, Heidelberg (2007)
11. Ron, K.: A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. In: Proceedings of the 14th International Joint Conference on Articial Intelligence (IJCAI 1995), pp. 1137–1145 (1995)
12. Robin, B.: Hybrid Web Recommender Systems. The Adaptive Web, 377–408 (2007)
13. Wong, S.K.M., Cai, Y.J., Yao, Y.Y.: Computation of Term Associations by a Neural Network. In: Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 1993), pp. 107–115 (1993)
14. Xu, Y.F., Zhang, L.: Personalized Information Service Based on Social Bookmarking. Implementing Strategies and Sharing Experiences. In: Proceedings of the 8th International Conference on Asian Digital Libraries, pp. 475–476 (2005)
15. Xu, Y., Zhang, L., Liu, W.: Cubic Analysis of Social Bookmarking for Personalized Recommendation. In: Zhou, X., Li, J., Shen, H.T., Kitsuregawa, M., Zhang, Y. (eds.) APWeb 2006. LNCS, vol. 3841, pp. 733–738. Springer, Heidelberg (2006)
16. Kerstin, B., Claudiu, S., Wolfgang, N., Raluca, P.: Can all tags be used for search? In: Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM 2008), pp. 193–202 (2008)
17. Guan, Z.Y., Bu, J.J., Mei, Q.Z., Wang, C.: Personalized Tag Recommenation Using Graph-based Ranking on Multi-Type Inerrelated Objects. In: Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2009), pp. 540–547 (2009)
18. Zhang, Z.K., Zhou, T., Zhang, Y.C.: Tag-Aware Recommender Systems: A State-of-the-Art Survey. Journal of Computer Science and Technology 26(5), 767–777 (2011)