International Journal of Modern Physics C
© World Scientific Publishing Company

EFFECTS OF USER TASTES ON PERSONALIZED RECOMMENDATION

JIAN-GUO LIU, TAO ZHOU, BING-HONG WANG, YI-CHENG ZHANG
Research Center of Complex Systems Science, University of Shanghai for Science and Technology, Shanghai 200093, P.R. China
Department of Modern Physics, University of Science and Technology of China, 230026, P.R. China
Department of Physics, University of Fribourg, Fribourg CH-1700, Switzerland
liujg004@ustc.edu.cn, zhutou@ustc.edu

QIANG GUO
School of Science, University of Shanghai for Science and Technology, Shanghai 200093, P.R. China

Received Day Month Year
Revised Day Month Year

In this paper, based on a weighted projection of the user-object bipartite network, we study the effects of user tastes on the mass-diffusion-based personalized recommendation algorithm, where a user's tastes or interests are defined by the average degree of the objects he has collected. We argue that the initial recommendation power located on the objects should be determined by both of their degree and the users' tastes. By introducing a tunable parameter, the user taste effects on the configuration of initial recommendation power distribution are investigated. The numerical results indicate that the presented algorithm could improve the accuracy, measured by the average ranking score, more importantly, we find that when the data is sparse, the algorithm should give more recommendation power to the objects whose degrees are close to the users' tastes, while when the data becomes dense, it should assign more power on the objects whose degrees are significantly different from user's tastes.

Keywords: Recommendation systems; Bipartite network; Network-based recommendation

PACS Nos.: 89.75.Hc, 87.23.Ge, 05.70.Ln

1. Introduction

With the rapid growth of the Internet and the World-Wide-Web, a huge amount of data and resource confront people with an information overload [1]. There are thousands of movies, millions of books, and billions of web pages on the web sites, and the amount of information is increasing more quickly than our personal processing abilities. This brings about massive amount of accessible information, which may result in a dilemma problem. It's hard for us to effectively filter out the pieces of information that are most appropriate for us. A landmark for information filtering is the use of search engine [2,3], by which user could find the relevant web pages by putting certain keywords. However, the search engine only returns the same results regardless of the users' tastes and interests.

Thus far, the most promising way to efficiently filter out the information overload is to provide *personalized recommendations*, which attempts to find out objects likely to be interesting to the target users by extracting the hidden information from the users' historical selections or collections. Motivated by its significance for economy and society, the design of efficient recommendation algorithms has become a common focus for computer science, mathematics, marketing practices, management science and physics. Various kinds of algorithms have been proposed, such collaborative filtering (CF) approaches [4,5,6,7,8], content-based analyses [9,10], network-based algorithm [11,12,13,14], hybrid algorithms [16,17], and so on. For a review of current progress, see Refs. [18,19] and the references therein.

Very recently, some physical dynamics, including mass diffusion (MD) [13,14] and heat conduction (HC) [11], have found their applications in personalized recommendations. These algorithms have been demonstrated to be of both high accuracy and low computational complexity [11,12,13,14]. Since MD and HC algorithms could be implemented based on the user-object bipartite network, it's also called *network-based algorithm*. The network-based algorithm supposes that the objects one user has collected have the power to recommend new objects to the target user, which is coincidence with the definition reachability [15]. In this paper, we introduce an improved MD algorithm with user-taste-dependent initial configuration. Compared with the uniform initial configuration, the prediction accuracy can be enhanced by using the user-taste-dependent configuration. More significantly, besides the prediction accuracy, we find that the data sparsity is an important factor affecting

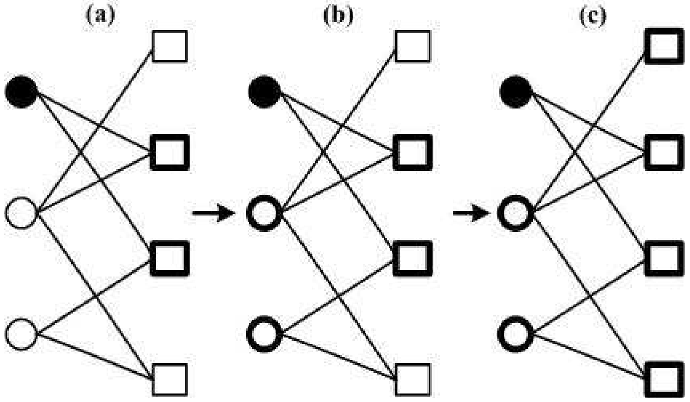

Fig. 1. Illustration of the network-based algorithm. The network-based algorithm could be applied in the following way. (a) The objects collected by the target user are activated; (b) The heat is diffused from the activated objects to the users who have collected them; (c) Then it's diffused back from the users to the objects.

the algorithm performance. When the sparsity of the user-object bipartite network is small, in other words, there are few edges between the users and objects, the algorithm should pay more attention on the users' habits and tastes, while when the number of edges in the bipartite network is large, the algorithm should give more recommendation power on the objects whose degrees significantly different with users' habits. Numerical simulations show that the improved algorithm has higher accuracy and can provide more diverse and less popular recommendations.

2. Mass-diffusion-based personal recommendation

In a recommender system, each user has voted or collected some objects, the system could be described by a bipartite network, in which there are two kind of nodes users and objects, the users' historical collection or selection behaviors could be well demonstrated by the edges connecting the users and objects. Formally, denote the object set as $O = \{o_1, o_2, \cdots, o_m\}$ and the user set as $U = \{u_1, u_2, \cdots, u_n\}$, the system can be fully described by a bipartite network with $m + n$ nodes, where there is an edge between a user and object if and only if this object is collected by the user. The bipartite network could be described by an adjacent matrix $\mathbf{A} = \{a_{ij}\} \in R^{m,n}$, where $a_{ij} = 1$ if o_i is collected by u_j, and $a_{ij} = 0$ otherwise. In MD algorithm, an object-object similarity network $\mathbf{W} = \{w_{\alpha\beta}\}_{m,m}$ is constructed firstly, where each node represents an object and two objects are connected if they have been collected simultaneously by at least one user. Then, to a target user, an amount of recommendation power is set on each object he has collected, and the proportion of the resource $w_{\alpha\beta}$ would like to distribute from o_β to o_α. In MD, a reasonable assumption is that the objects that users have collected are what they like, and the objects a target user has collected would be regarded as the initial mass source, then the activated objects redistribute the mass to the users who have collected them before, with users receiving a level of mass equal to the mean amount possessed by their neighboring objects, and objects then receiving back the mean of their neighboring users mass levels. Due to the sparsity of real data sets, these "physical" descriptions of the algorithm turn out to be more computationally efficient in practice than constructing and using the object similarity matrix $\mathbf{W}$, and MD algorithm could be implemented in three steps on the user-object bipartite network, which is shown in Fig.1(a-c).

Lind *et. al.* presented a cycle measurement to investigate the clustering property in bipartite network [20,21]. According to the algorithm description and the cycle definition, the object similarity of the mass-diffusion-based algorithm can be expressed as [13],

$$w_{\alpha\beta} = \frac{1}{k(o_\beta)} \sum_{l=1}^{n} \frac{a_{\alpha l} a_{\beta l}}{k(u_l)}, \tag{1}$$

where $k(o_\beta) = \sum_{i=1}^{n} a_{\beta i}$ and $k(u_l) = \sum_{i=1}^{m} a_{il}$ denote the degrees of object o_β and user u_l, respectively. For a target user, in the simplest case, the initial resource

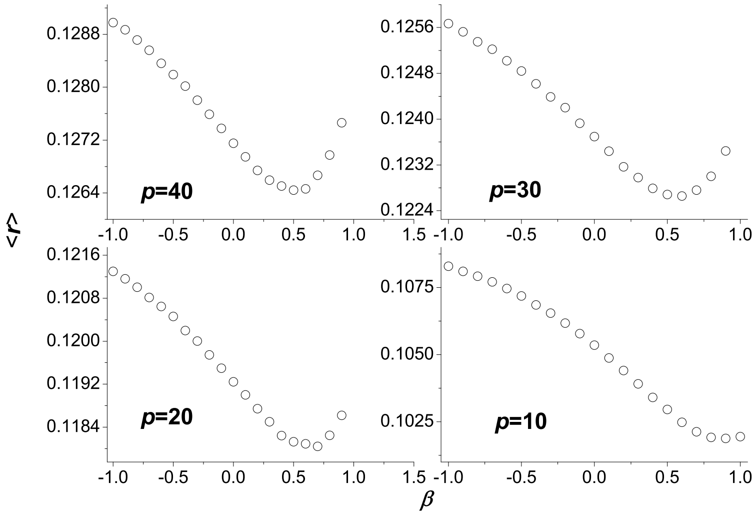

Fig. 2. Average ranking score $\langle r \rangle$ vs. β when $p = 10, 20, 30$ and 40. All the data points are averaged over ten independent runs with different data-set divisions.

vector $\mathbf{f} = \{f_1, f_2, \cdots, f_m\}^T$ can be set as

$$f_j = a_{ji}. \tag{2}$$

In other words, only the objects user u_i has collected are set unit resource. After the mass-diffusion-process demonstrated in Fig.1, the final resource vector is

$$\widehat{\mathbf{f}} = W\mathbf{f}. \tag{3}$$

Sorting the vector $\widehat{\mathbf{f}}$ in descending order according to value of $\widehat{f_j}$, the objects obtained highest values are recommended to the target user.

3. Improved algorithm by considering the user taste effects

In the standard MD algorithm, for any user u_i, all of the collected objects are assigned the same recommendation power. Although it already has a good algorithmic accuracy, this uniform configuration may be oversimplified, and didn't consider the effects of user tastes. In this paper, the user taste is defined by the average object degree he has collected. The objects whose degrees are close to the user taste should be assigned more recommendation power. We also notice that most of the user tastes are less than 100, while the degrees of the popular objects are close to 300. If the recommendation power is assigned according to the distance between the object degree and the user taste, it will give more power on the popular objects and weaken the unpopular object effects. In order to balance the objects whose degrees

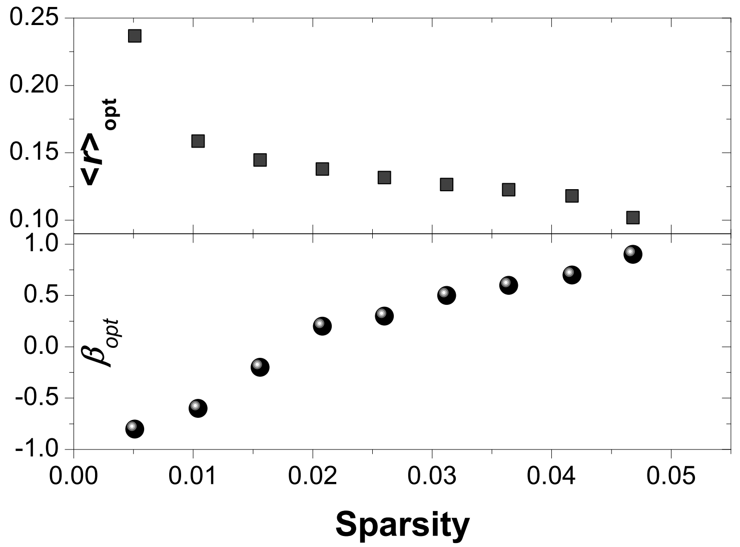

Fig. 3. The optimal β_{opt} and the corresponding average ranking score $\langle r\rangle_{\mathrm{opt}}$ vs. the sparsity of the training set. All the data points are averaged over ten independent runs with different data-set divisions.

are larger or less than the user tastes, we present a more complicated distribution of initial resource according to the following way.

$$f_\alpha^i = a_{\alpha i} I_{\alpha i}, \tag{4}$$

where $I_{\alpha i}$ is defined as follows

$$I_{\alpha i} = \begin{cases} (k(o_\alpha)/\overline{k}(u_i))^\beta & k(o_\alpha) \geq \overline{k}(u_i) \\ (\overline{k}(u_i)/k(o_\alpha))^\beta & k(o_\alpha) < \overline{k}(u_i) \end{cases} \tag{5}$$

where $\overline{k}(u_i)$ denote the average degree of user u_i's collected objects, and β is a tunable parameter. Compared with the uniform case, $\beta = 0$, a positive β strengthens the influence of the objects whose degrees are larger or less than $\overline{k}(u_i)$, while a negative β strengthen the influence of the objects whose degrees are close to $\overline{k}(u_i)$.

4. Numerical results

A benchmark dataset, namely MovieLens, is used to test the improved algorithm. The MovieLens data is a randomly-selected subset of the huge data, which consists of 1682 movies (objects) and 943 users. The users vote movies by discrete ratings from one to five. We applied a coarse-graining method: A movie is set to be collected by a user only if the giving rating is larger than 2. The original data contains 10^5 ratings, 85.25% of which are ≥ 3, that is, the user-object (user-movie) bipartite network after the coarse gaining contains 85250 edges. We randomly divide this data set into two parts: one is the training set, treated as known information, and the other is the probe, whose information is not allowed to be used for prediction. We use a parameter p to control the data density, that is, $p\%$ of the ratings are put into the probe set, and the remains compose the training set.

A good recommender method should rank preferable objects to math the users' tastes. Therefore, the collected objects in the probe set should be set at the top level of the recommendation lists. The average ranking score is adopted to measure the accuracy. It could be defined as follows. For a target user u_i, if the entry u_i-o_j is in the probe set, we measure the position of o_j in the ordered list. For example, if there are $L_i = 10$ uncollected objects for u_i, and o_j is the 2nd one from the top, we say the position of o_j is $2/10$, denoted by $r_{ij} = 0.2$. A good algorithm is expected to give high recommendations to them, thus leading to small r_{ij}. Therefore, the mean value of the position $\langle r\rangle$ can be used to evaluate the algorithmic accuracy: the smaller the average ranking score, the higher the algorithmic accuracy, and vice verse. The average degree of all recommended objects, $\langle k\rangle$, and the mean value of Hamming distance, S, are taken into account to evaluate the popularity and diversity. The smaller average degree, corresponding to the unpopular objects, are preferred since those small-degree objects are hard to be found by users themselves. The diversity can be quantified by the average Hamming distance, $S = \langle H_{ij}\rangle$, where $H_{ij} = 1 - Q_{ij}/L$, L is the length of recommendation list and Q_{ij} is the overlapped number of objects in u_i and u_j's recommendation lists. The largest $S = 1$ indicates the recommendations to all of the users are totally different, while the smallest $S = 0$ means all of recommendations are exactly same.

Implementing the improved algorithm on the MovieLens data, the accuracy, popularity and diversity are investigated. Figure 2 reports the algorithmic accuracy as a function of β to different p, from which one can find that the curves obtained by the improved algorithm has clear minimums, which strongly support the above discussion. Compared with the routine case ($\beta = 0$), the average ranking score can be reduced 5.6% at the optimal case when $p = 10$. Numerical results on different percentage of probe sets show that the optimal parameter $\beta_{\rm opt}$ decreases with the increase of p. Figure 3 reports the relation between the optimal $\beta_{\rm opt}$, the corresponding average ranking scores $\langle r\rangle_{\rm opt}$ and the sparsity of the training sets. One can see from Fig.3 that the optimal $\langle r\rangle_{\rm opt}$ is negatively correlated with the data sparsity, where the sparsity is defined as $\frac{E}{m\times n}$, and E is the number of edges in the user-object bipartite network, more interestingly, the optimal parameter $\beta_{\rm opt}$ is positively correlated with the sparsity. The reason may lie in the fact that when the users have not collected too much objects, their tastes are easy to be distinguished, therefore, the objects whose degrees are close to $\overline{k}(u_i)$ should be assigned more recommendation power. As the number of users' collected objects increases, users' tastes become diversity, therefore, it's hard to catch the users interested and habits. Under these circumstances, the users are more interesting to the objects different from his historical collects which could bring him/her fresh information.

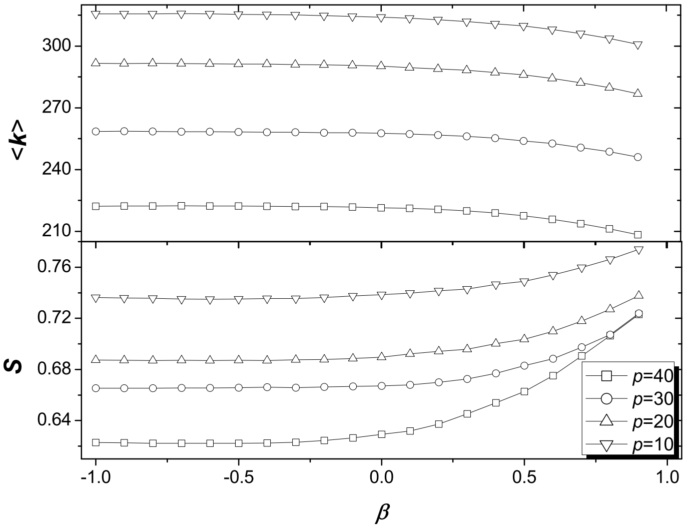

Fig. 4. When the recommendation list $L = 10$, $\langle k \rangle$ and S vs. β of $p = 10, 20, 30$ and 40. All the data points are averaged over ten independent runs with different data-set divisions.

Besides accuracy, the popularity and diversity are also investigated. Figure 4 reports the average degree and diversity of all recommended movies as a function of β to different p when the recommendation lists $L = 10$, from which one can find that although the average object degrees scarcely change, the diversity is increased at the optimal $\beta_{\rm opt}$.

5. Conclusion and Discussion

In this paper, the effects of user tastes on MD recommendation algorithm are investigated, where the user tastes are defined by the average object degree he/her has collected. By introducing a free parameter β, an improved algorithm by regulating the initial configuration of resource is presented. Numerical results indicate that when the data set is sparse, it's easy to distinguish the users' tastes and the objects whose degrees are close to the users' tastes should be assigned more recommendation power, while as the data set becoming dense, the objects whose degree far from the users' tastes should be emphasized. Besides the average ranking score, the popularity and personalization of recommended objects are also taken into account. The results show that the improved algorithm outperforms the standard MD algorithm in both of accuracy and personalization.

In the improved algorithm, we only give a kind of user taste definition, however, there are several other ways to define the users' tastes, such as time-dependent behavior, variance of the user collected object degrees, and so on. We believe MD algorithm could be further improved by catching the users' current tastes.

Instead of calculating all the elements in $\mathbf{W}$, one can implement the current algorithm by directly diffusing the resource of each user. Ignoring the degree-degree correlation in user-object relations, the algorithmic complexity is $O(m\langle k_u\rangle\langle k_o\rangle)$, where $\langle k_u\rangle$ and $\langle k_o\rangle$ denote the average degrees of users and objects. Theoretical physics provides us some beautiful and powerful tools in dealing with this long-standing challenge in modern information science: how to do a personal recommendation. The presented algorithm also could be used to find the relevant reviewers for the scientific papers or funding applications [22,23], and the link prediction in social and biological networks[24]. We believe the current work can enlighten readers in this promising direction.

This work is partially supported by the National Basic Research Program of China (No. 2006CB705500), the National Natural Science Foundation of China (Nos. 60744003, 10635040, 10472116), the Swiss National Science Foundation (Project 205120-113842), and Shanghai Leading Discipline Project(Grant No. S30501).

References

1. G.-Q. Zhang, G.-Q. Zhang, Q.-F. Yang, S.-Q. Cheng and T. Zhou, *New Journal of Physics* **10**, 12307 (2008).
2. S. Brin and L. Page, *Comput. Netw. ISDN Syst.* **30**, 107 (1998).
3. J.M. Kleinberg, *J. ACM* **46**, 604 (1999).
4. J.L. Herlocker, J.A. Konstan, K. Terveen and J. Riedl, *ACM Trans. Inform. Syst.* **22**, 5 (2004).
5. J.A. Konstan, B.N. Miller, D. Maltz, J.L. Herlocker, L.R. Gordon and J. Riedl, *Commun. ACM* **40**, 77 (1997).
6. J.-G. Liu, B.-H. Wang and Q. Guo, *Int. J. Mod. Phys. C* **20**, 285 (2009).
7. R.-R. Liu, C.-X. Jia, T. Zhou, D. Sun and B.-H. Wang, *Physica A* **388**, 462 (2009).
8. D. Sun, T. Zhou, J.-G. Liu, R.-R. Liu, C.-X. Jia and B.-H. Wang, *Phys. Rev. E* **80**, 017101 (2009).
9. M. Balabanović and Y. Shoham, *Commun. ACM* **40**, 66 (1997).
10. M.J. Pazzani, *Artif. Intell. Rev.* **13**, 393 (1999).
11. Y.-C. Zhang, M. Blattner and Y.-K. Yu, *Phys. Rev. Lett.* **99**, 154301 (2007).
12. Y.-C. Zhang, M. Medo, J. Ren, T. Zhou, T. Li and F. Yang, *Europhys. Lett.* **80**, 68003 (2008).
13. T. Zhou, J. Ren, M. Medo and Y.-C. Zhang, *Phys. Rev. E* **76**, 046115 (2007).
14. T. Zhou, L.-L. Jiang, R.-Q. Su and Y.-C. Zhang, *Europhys. Lett.* **81**, 58004 (2008).
15. P.G. Lind, L.R. da Silva, J.S. Andrade and H.J. Herrmann, *Phys. Rev. E* **76**, 036117 (2007).
16. N. Good, J.B. Schafer, J.A. Konstan, A.l. Borchers, B. Sarwar, J. Herlocker and J. Riedl, *Proc. Conf. Am. Assoc. Artificial Intelligence*, 439 (1999).
17. M. Pazzani and D. Billsus, *Machine Learning* **27**, 313 (1997).
18. G. Adomavicius and A. Tuzhilin, *IEEE Trans. Know. & Data Eng.* **17**, 734 (2005).
19. J.-G. Liu, M.Z.Q. Chen, J. Chen, F. Deng, H.-T. Zhang, Z. Zhang and T. Zhou. *Int. J. Info. and Syst. Sci.* **5**(2), 230 (2009).
20. P.G. Lind, M.C. González and H.J. Herrmann, *Phys. Rev. E* **72**, 056127 (2005).
21. P.G. Lind, H.J. Herrmann, *New Journal of Physics* **9**, 228 (2007).

22. J.-G. Liu, Y.-Z. Dang and Z.-T. Wang, *Physica A* **366**, 578 (2006).
23. J.-G. Liu, Z.-G. Xuan, Y.-Z. Dang, Q. Guo and Z.-T. Wang, *Physica A* **377**, 302 (2007).
24. T. Zhou, L. Lv and Y.-C. Zhang, arXiv:0901.0553v1.