INTERNATIONAL JOURNAL OF
INFORMATION AND SYSTEMS SCIENCES
Volume 5, Number 2, Pages 230–247



# RECENT ADVANCES IN PERSONAL RECOMMENDER SYSTEMS

JIAN-GUO LIU, MICHAEL Z. Q. CHEN, JIANCHI CHEN, FEI DENG, HAI-TAO ZHANG, ZI-KE ZHANG, AND TAO ZHOU

**Abstract.** In the past years we have witnessed an explosive growth of the data and information on the World Wide Web, which makes it difficult for normal users to find the information that they are interested in. On the other hand, the majority of the data and resources are very unpopular, which can be considered as "hidden information", and are very difficult to find. By building a bridge between the users and the objects and constructing their similarities, the Personal Recommender System (PRS) can recommend the objects that the users are potentially interested in. PRS plays an important role in not only social and economic life but also scientific analysis. The interdisciplinary PRS attracts attention from the communities of information science, computational mathematics, statistical physics, management science, and consumer behaviors, etc. In fact, PRS is one of the most efficient tools to solve the information overload problem. According to the recommendation algorithms, we introduce four typical systems, including the collaborating filtering system, the content-based system, the structure-based system, and the hybrid system. In addition, some improved algorithms are proposed to overcome the limitations of traditional systems. This review article may shed some light on the study of PRS from different backgrounds.

**Key Words.** Personal recommender system, Collaborative filtering, Content-Based analysis, Structure-Based algorithm, Hybrid systems, bipartite network, Infophysics.

## 1. Introduction

In just a few years, both the Internet and World Wide Web (WWW) have experienced an exponential growth. As a result, people's traditional perceptions on commercial interactions and social environment changed correspondingly. More and more customers attempt to buy goods through the Internet, while companies are looking for their customers online to increase sales, e.g., *Amazon.com*, the biggest bookshop on the Internet, and *Google.com*, the famous search engine. (See also the annual survey on behaviors of web users by *ChoiceStream*.) E-commerce has played an important role in many people's lives. However, facing possible selections

Received by the editors September 1, 2008 and, in revised form, March 12, 2009.
2000 *Mathematics Subject Classification.* 70E55, 70E60, 15A39.
The authors would like to thank Y. Du and S. Hassan for their help in preparing the manuscript. This work is partially supported by SBF (Switzerland) for financial support through project C05.0148 (Physics of Risk), the Swiss National Science Foundation (project 205120-113842), the National 863-Project No. 2007AA12Z154, the National Natural Science Foundation of China under grant No. 10635040, 60744003 and 60704041 and a funding from Liaoning Education Department (Grant No. 20060140).

among thousands of products, consumers must spend much time to search the information on products and evaluate them. On the other hand, many business practitioners try to seek the affinity in buyers' tastes. Compared with the past, the opportunities are much more easily exploitable due to the "Information Revolution" and advances of network interfaces. For instance, the recent technology WEB2.0 helps the business collect a large amount of information on buyers' preferences more easily due to the connections between people on an unprecedented scale. The web-users' feedbacks, comments, and their choices of products and services are collected by these websites, which play a significant role in the implementation of the business development. Accordingly, sellers can predict the customers' personal preferences and make personal recommendations. *Maslov* and *Zhang* [1] explained that people's consumption patterns are not based on the complete optimization over all choices. Instead, they generate "hidden wants", which makes much better matches between customers and products. For example, when a customer needs a new mobile, he or she can find the recommendation information on the mobiles which match his/her desired functions from the Internet. And the customer will consider other users' comments and their experiences of using the same cellular phone. Thus, the website system must solve the problem of how to make a recommendation and give sufficient information to the users. In the past few years, the websites emphasized that the relationships among the users play more and more important roles in the E-commerce sales. Popular products attract many customers, while other less popular ones aim at some particular groups of users. The objectives of the recommender systems are:

(1) Find out products which users are interested in and turn the Internet browsers into customers.
(2) Improve the sales of products on the e-commerce websites.
(3) Increase the customer loyalties towards the websites/products.

One of the key factors to measure the recommendation algorithm is the accuracy. If the recommendation lists made by the system is very different from the users' interests, the users will lose his/her confidence in the system and probably ignore the future information provided by system. The system makes recommendations based on the users' records, which should be as personalized as possible so that the users' requirements can be sufficiently satisfied.

With the development of web technology, users and companies need a recommender system to find out the most relevant information among the hundreds of millions alternatives [2, 3]. There are a lot of companies offering recommendation engines, which can provide products, services, and others information to users according to their personal tastes. According to a report of the *Yankee Group*, there is a 400% increase in click-through rate on some websites.

In [4], *Felfernig et al.* made a short introduction to the recommender system, including recommendation approaches and recommendation environment. Good *et al.* [5] compared several different recommender systems such as *Amazon.com*, *CD-NOW*, *eBay*, *Levis*, *Moviefinder.com*, and *Reel.com*. They proposed a set of suggestions for real applications of recommender systems. Generally speaking, there are four typical recommender systems: the collaborative filtering system, the content-based system, the structure-based system, and the hybrid method system.

- Collaborative recommendations: The user will be recommended objects that people with similar taste and preferences liked in the past;
- Content-based recommendations: The user will be recommended objects similar to the ones this user preferred in the past;

- Structured-based recommendations: Recommendations are given based on the user-object bipartite networks;
- Hybrid approaches: Integration of multiple methods.

The rest of this paper is organized as follows. Section 2 reviews the collaborative filtering system. Section 3 presents recent advances in content-based recommender systems. Structure-based recommendation algorithms are surveyed in Section 4. Section 5 summarizes the progress in the hybrid approaches. Finally, conclusion and discussion are given in Section 6.

## 2. Collaborative filtering system

The first generation of Personal Recommender System (PRS) is the collaborative filtering system (CFS), which has been widely applied in e-commerce systems since its invention. The underlying mechanism of CFR consists of two parts. First, the users' similarities are computed based on the users' past activities. Second, the opinions of users with high similarities to the target user are integrated to form recommendations. The most important advantage of CFS is that the system has no special requirement on the objects, therefore CFS can recommend the objects without clear content descriptions, e.g., music and movies.

CFS is the most widely applied PRS thus far, and *Grundy* [6] is regarded as the first application. *Grundy* first builds the user interest models, and recommends the relevant books to every user based on his/her models. *Tapestry* [7], the mail-distribution system, confirms the users' similarity manually. The computational complexity, however, would increase dramatically with the number of users, and the accuracy would decrease simultaneously. *GroupLens* [8] sets up the user community first, and the users may propagate their own opinions to other members within the same community. By using the social information filtering method, *Ringo* [9] recommends music to its users. There exist many other CFS applications, such as the book recommender system *Amazon.com* [10], joke recommender systems Jester [11], and WWW information recommendation system *PHOAKS* [12].

CFS can be classified into two categories: memory-based [13, 14, 15] and model-based systems [16, 17, 18, 19, 20]. The memory-based algorithms provide the predicted scores based on the information presented by all the marked objects. Let $C$ be the user set $C = \{c_1, c_2, \cdots, c_N\}$, $S$ be the object set $S = \{s_1, s_2, \cdots, s_M\}$, and $r_{c,s}$ be the predicted score from user $c$ to an uncollected object $s$. In CFS, the predicted scores are computed based on preferences of the target users' neighbors (here, the neighbors refer to a set of users with highest similarities to the target user). Let $\hat{C}$ be the set of neighbors of user $c$, and the predicted score $r_{c,s}$ can be obtained as:

$$(1) \qquad \begin{array}{ll} (a) & r_{c,s} = \frac{1}{N}\sum_{\overline{c}\in\hat{C}} r_{\overline{c},s}, \\ (b) & r_{c,s} = k\sum_{\overline{c}\in\hat{C}} \mathrm{sim}(c,\overline{c}) \times r_{\overline{c},s} \\ (c) & r_{c,s} = \overline{r}_c + k\sum_{\overline{c}\in\hat{C}} \mathrm{sim}(c,\overline{c}) \times (r_{\overline{c},s} - \overline{r}_c), \end{array}$$

where $k$ is the normalization factor, defined as $k = 1/\sum_{\overline{c}\in\hat{C}} |\mathrm{sim}(c,\overline{c})|$, and $\mathrm{sim}(i,j)$ denotes the similarity between user $i$ and $j$. The average score $\overline{r}_c$ is defined as $\overline{r}_c = \frac{1}{|s_c|}\sum_{s\in S_c} r_{c,s}$ with $S_c = \{s \in S | r_{c,s} \neq 0\}$. As shown in Eq. (1a), the simplest way is to average all scores given by the neighboring users. Instead, formula (1b) is the most extensively used one, where the weight $\mathrm{sim}(c,\overline{c})$ is added, which means that the scores given by the users with higher impacts on the final predicted scores. Although the weighted sum is used in (1b), in real systems, the scores given by different users tend to have different meanings. Eq. (1c) overcomes this problem by considering the diversity of rating criteria of different users. Generally speaking, (1c) is more accurate than (1b). The preferential filtering [21, 22, 23, 24] is another important method to overcome this deficiency, which focuses on predicting the relevant preferences instead of the absolute scores.

Many algorithms have been implemented in CFS to compute the user similarities [2, 9, 13, 25, 26, 27, 28], with the majority based on the scores given by different users to the same objects. Two commonly used algorithms are Pearson coefficient and cosine similarity, both of which are based on the common rated object set $S_{xy} = S_x \cap S_y$ obtained directly by graph theory [29]. The Pearson coefficient between users $x$ and $y$ is defined as [2, 9]

$$\mathrm{sim}(x,y) = \frac{\sum_{s\in S_{xy}}(r_{x,s}-\overline{r}_x^2)(r_{y,s}-\overline{r}_y^2)}{\sqrt{\sum_{s\in S_{xy}}(r_{x,s}-\overline{r}_x^2)\sum_{s\in S_{xy}}(r_{y,s}-\overline{r}_y^2)}}. \tag{2}$$

In the cosine similarity [13, 25], every user is first represented as an $m$-dimensional vector, then the similarity between two vectors can be measured by the cosine function as follows:

$$\mathrm{sim}(x,y) = \cos(\overrightarrow{x},\overrightarrow{y}) = \frac{\overrightarrow{x}\cdot\overrightarrow{y}}{\|\overrightarrow{x}\|_2 \times \|\overrightarrow{y}\|_2}, \tag{3}$$

where $\overrightarrow{x}\cdot\overrightarrow{y}$ denotes the dot product of two vectors. In order to increase the prediction accuracy, different methods have been used to compute the user similarities. Since the users' interests and habits vary with time, a widespread strategy is to compute the user similarities beforehand and update them periodically.

Many variants of the Pearson coefficient and cosine similarity methods have been proposed and extensively studied, e.g., the default voting, case amplification, weighted advantage prediction, etc. The default voting [13] is an extension to the memory-based method. If the number of rated objects is very small, the accuracy obtained by all the above methods would not be satisfactory because of the fact that these methods are based on the common rated object set $S_{xy}$. Numerical results indicate that, if a default score is allocated to every uncollected object, the prediction accuracy would be increased. Sarwar *et al.* [25] presented a new method by combining the Pearson coefficient and cosine formula to compute the object similarities. *Deshpande* and *Karypis* [30] reduced this method to the top-$N$ recommendation algorithm, in which only the most similar $N$ objects are considered for recommendation. The numerical simulation results show that this algorithm is not only more accurate than the previous ones, but also one or two magnitudes faster than the previous ones. *Chen* and *Cheng* [27] computed the user similarities based on the object's order on the user-object list, in which the objects in the top position carry heavier weight to compute the user similarities. In addition, *Yang* and *Gu* [28] presented a method to construct the user interest model according to the user's historical behaviors. Numerical results indicate that this method is better than the standard collaborative filtering algorithm. The results in [25, 30] show that the recommendation algorithms based on the object similarities outperform the ones based on the user similarities.

In the model-based algorithms, the system gives the predicted scores by collecting the user marked data and constructing the user behavior model. *Breese et al.* [13] presented a collaborative filtering algorithm based on a statistical model. Assume that the marked scores are integer, then the predicted score could be given by

$$r_{c,s} = E(r_{c,s} = \sum_{i=0}^{n} \times \Pr(r_{c,s} = i) | r_{c,\overline{s}\overline{s}} \in S_c). \tag{4}$$

In order to estimate the statistical value, *Breese et al.* [13] presented two selective statistical models: clustering model and the Bayesian Networks. In the former, the user marks are independent, and the users with the same interests are clustered together. In the latter, the number of clusters and the model parameters can be directly obtained from the existing data.

There are many CFSs, such as the statistic relevance algorithm [17], maximum entropy method [20], linear regression method [25], Gibbs sampling based on clustering [31], Bayesian model [32], latent semantic analysis [18, 33, 34], and aspect model [18]. Recently, most of the work attempted to construct the recommendation model by using more complex statistical models. For example, *Shani et al.* [35] viewed the recommendation process as a decision-making process based on the Markov theory. In addition, by using a trivial statistical model, *Kumar et al.* [36] demonstrated that the small-size data set is very important in CFS. *Yu et al.* [37] presented some modified methods from the perspective of data processing, including the noise removal, user marked data selecting, redundancy analysis, sparse data processing, and so on. The simulation results showed that these methods can improve the recommendation accuracy and efficiency. *Manouselis* and *Costopoulou* [38] presented a multi-criteria CFS, which could recommend the objects with multiple criteria. *Chen et al.* [39] presented the community CFS, which can recommend a community instead of merely one object.

In summary, the CFS has the following advantages.

**1:** CFS can recommend new information to the users, even if the interests have not been discovered by the user himself.

**2:** CFS can recommend the objects that are hard to express structurally, such as art, music, and so on.

Although CFS has been used extensively, it also has some disadvantages, such as the cold-start problem (how to recommend to the new users who have no prior comments or marks and how to evaluate the new objects), mark sparsity, the extendibility, and so on. In addition, the computational complexity increases linearly with the number of the users. Therefore, considering a system with almost a fixed number of objects, the object similarities are very important and the algorithm based on the object similarity is very efficient. Since the response speed is one of the most important factors to the users, more and more systems begin to adopt the object-based recommendation algorithm.

## 3. Content-based system

The initial content-based recommendation (CBR) is a derivation of CFS, which gives the predicted scores based on the object content information instead of the users' comments or marks. Based on the machine learning technology, CBR could construct the profiles of users and objects respectively, and update the user profiles periodically. The system could also compare the profiles between the users and objects, then give the recommendation results directly. For example, in the movie recommender system, CBR first analyzes the common characteristics (actors, directors, styles, etc.) of the movies watched by a target user, then recommends the movies with the same features. Basically, CBR relies on the information acquisition [40, 41] and filtering technologies [42]. With the maturity in text information acquisition and filtering technologies, it is easy to analyze the text information of the objects, which may be beneficial to the development of CBR.

In the information acquisition process, TF-IDF [41] is one of the most widely used algorithms to express the text content. More precisely, assume that there are $N$ text files, the keyword $k_i$ appears in $n_i$ files, let $f_{ij}$ be the number of appearances of keyword $k_i$ in the file $d_j$, then the frequency $TF_{ij}$ is defined as

$$TF_{ij} = \frac{f_{ij}}{\max_z f_{zj}}. \tag{5}$$

In many text files, the keywords emerging in many files at the same time contribute little to distinguish the file relevance. Therefore, $TF_{ij}$ is always used with the inverse number of the emerging times $IDF_i$, defined as

$$IDF_i = \log \frac{N}{n_i}. \tag{6}$$

A file $d_j$ could be expressed by the vector $d_j = (w_{1j}, w_{2j}, \cdots, w_{kj})$, with

$$w_{ij} = \frac{f_{ij}}{\max_z f_{zj}} \log \frac{N}{n_i}. \tag{7}$$

Denote Content($s$) by the profile of the object $s$, which is composed of some descriptive words. Generally, Content($s$) could be extracted from the object description. In most CBR systems, the object contents are described by the keywords. A typical example is the *Fab* system [43]. Fab, the web recommender system, expresses a web page by its most important 100 keywords. *Pazzani* and *Billsus* [44] expresses a file by using the most 128 informative keywords. CBR gives the recommendation results based on a user's previous selections, and compares the similarities between the connected objects and the unconnected ones, then recommends the most similar objects to the user. Alternatively, it compares the similarities between a user profile and the unconnected object profiles directly. Denote the user profile by $c$, which can also be expressed as a vector $(w_{c1}, w_{c2}, \cdots, w_{ck})$, where the weight $w_{ci}$ denotes the importance of the keyword $k_i$ to user $c$. Thus, both of the users and objects can be expressed in the TF-IDF formulation. In CBR, $r_{c,s}$ is defined as

$$r_{c,s} = \mathrm{score}(\mathrm{UserProfile}(c), \mathrm{Content}(s)). \tag{8}$$

$r_{c,s}$ can be obtained by the vectors $\overrightarrow{w}_c$ and $\overrightarrow{w}_s$. For example, based on the cosine formula [40,41]:

$$r_{c,s} = \cos(\overrightarrow{w}_c, \overrightarrow{w}_s) = \frac{\overrightarrow{w}_c \cdot \overrightarrow{w}_s}{\|\overrightarrow{w}_c\|_2 \times \|\overrightarrow{w}_s\|_2}. \tag{9}$$

Apart from the classical algorithms based on information acquisition, some real systems have adopted other technologies, such as Bayesian clustering [44, 45, 46], clustering analysis, decision tree, neural networks [45], and so on. The difference between these algorithms and the information acquisition lies on the fact that they are not based on a function or formulation, but based on the statistical information and machine learning.

In CBR, one of the key points is to update the profiles of users and objects. Recently, *Somlo* and *Howe* [47] and *Zhang et al.* [48] presented a new updating method by using the adaptive filtering technology. First, the model constructs the user profile based on the user's interests and habits recorded in some subject files afterwards. Then, it compares the file similarities between the content of a web file database and the subject files, and selects the webs having high similarities with the user profile to update its profiles. In addition, based on the adaptive filtering technology, Robertson and Walker [49] and *Zhang et al.* [50] developed an optimal matching threshold algorithm, which is detailed as follows. First, it finds an optimal threshold value according to the existing data and the similarity distribution of the user profiles, and distinguish whether a file is relevant to a user profile. Only the files with similarities higher than the threshold could be used to update user profiles. This method not only increases the accuracy but also improves the system efficiency.

Basically, a user profile consists of many keywords and hence we could save the storage space by using the graph index method. The system, however, must spend much of its resources to update these profiles when the user interests and habits change. Chang *et al.* [51] solved this problem by introducing both long-time and short-time interests. In their method, the short-time interests are assigned more weight, thereby the updating tree is reconstructed, which reduces the cost of updating the user profiles. Compared with the classical keyword-based method, Degemmis, which is based on the WordNet [52], constructed the semiology user profile by using the machine learning and text clustering algorithms. In the semiology, the profile consists of the sematic information of the user interests, not only keywords. The numerical results on CBR indicate that this method could enhance the recommendation accuracy. AdROSA advertisement recommender system [53] constructed the profiles by using the user registration information and the IP address, browsing habits etc. The profiles are used to compare with the Web content, and only the websites with high similarities would be recommended.

Although the algorithms are capable of automatically acquiring and updating the user profiles, it must keep the balance between the accuracy and speed simultaneously. The algorithms, which catch the true user information well, are not easy to update. Conversely, if we enhance the update speed, the accuracy would decrease as a result. Human-computer interaction is one of the most efficient remedies. *Ricci* and *Nguyen* [54] designed online a mobile traveling recommender system, which could obtain the user demand based on a simple human-computer interaction and then recommend the satisfactory travel routine or products to the users. At the beginning, even the users are not well aware of their interests, but after some interactive inquiries, both the users and system would acquire it.

Another big challenge to CBR is that the profiles in different languages are not compatible. *Martínez et al.* [55] presented a flexible language method, which could express the user profiles by many languages, then these profiles could be used in different language systems.

In summary, CBR has the following advantages:

**1:** CBR can deal with the cold-start problem. Without available information on the new users and objects, CFR cannot evaluate the new objects and recommend to the new users. CBR, however, can handle it by using the profiles.

**2:** In some real systems, the marks are very sparse. Therefore, the recommender systems depending on the user's previous marks are ineffective. On the other hand, CBR, which is based on the content profiles, is still effective.

**3:** CBR can recommend new and unpopular objects, in other words, finding the "hidden" information.

**4:** By demonstrating the content properties of the recommended objects, CBR can explain to the users why these objects are recommended.

Inevitably, CBR is constrained by the development of the information acquisitions, e.g., automatically extracting the content from the multimedia data (such as pictures, videos, music, and so on), which have prevented the development or success of CBR. In the next section, we will introduce the structure-based recommender system, which is based on the user-object bipartite network. This kind of algorithm is not restricted by the information mining technologies, and can solve the sparsity and extendibility problems.

## 4. Structure-based system

In the structure-based system, both the user and object are represented as a node. The user-object correlations contain all the information needed by the algorithms. *Zhou et al.* [56, 57] and *Huang et al.* [58, 59] constructed the user-object bipartite network and set up the correlations between the users and objects, based on which they presented the structure-based recommendation (SBR) algorithm respectively. By using the mass diffusion processing, *Zhou et al.* [56, 57] presented the mass-diffusion recommendation algorithm. By introducing the spreading activation into the collaborative filtering algorithms, *Huang et al.* [59] partially solved the sparsity problem. Furthermore, they [59] analyzed two user-object bipartite networks in real systems, and found that both of them have large average distance and clustering coefficient than the ones obtained by random networks. Considering the marked objects, by introducing the physics theory into the PRS, *Zhang et al.* [60, 61] studied the heat conduction process on the user-object bipartite network. Recently, *Zhou et al.* [62] presented a hybrid algorithm by combing the mass diffusion and heat conduction algorithms, resulting in higher accuracy. All the above algorithms outperform the classical collaborative filtering algorithms (see also Section 2). The progress of the structure-based algorithm will be introduced in this section.

**4.1. How to evaluate the algorithm's performance?** To test the algorithm's accuracy, the data set is divided into two parts: one is the training set used as known information for prediction, and the other one is the probe set, whose information is not allowed to be used for prediction. Many indexing methods are proposed to evaluate the algorithm's accuracy, including *precision*, *recall*, *F-measure*, *ranking score* [56], and so on. Indeed, a recommendation algorithm should provide each user with an ordered list of all its uncollected objects. For an arbitrary user $u_i$, if the relation $u_i$-$o_j$ is in the probe set (according to the training set, $o_j$ is an uncollected object for $u_i$), the position of $o_j$ in the ordered list is measured. For example, if there are $L_i = 100$ uncollected objects for $u_i$, and $o_j$ is the 10th from the top, we say that the position of $o_j$ is 10/100, denoted by $r_{ij} = 0.1$. Since the probe entries are actually collected by users, a good algorithm is expected to give high recommendations to them, thus leading to small $r_{ij}$. Therefore, the mean value of the position $r_{ij}$, or $\langle r \rangle$ (called *ranking score* [56]), averages over all the entries in the probe, can be used to evaluate accuracy of the algorithm: the smaller the ranking score, the higher the accuracy, and vice versa.

**4.2. Bipartite network and the recommendation algorithms.** Denote the number of users by $m$ and let $n$ be the one of the objects in a recommender system. If a user $i$ has collected object $j$, there is one edge between $i$ and $j$, denoted by $a_{ji} = 1$; otherwise, $a_{ji} = 0$. In this way, a recommender system could be represented by a bipartite network with $m + n$ nodes. For each user, the final aim of the recommender system is to sort the uncollected objects for the users. All the objects collected by user $i$ have some ability to recommend other objects to this user. Such an abstract ability could be regarded as resource. The objects prefer to give more resources to other objects of high similarity. Consider a recommender system with $m$ users and $n$ objects, and the quota given to object $i$ from object $j$, denoted by $w_{ij}$, is defined as [56]

$$w_{ij} = \frac{1}{k_i} \sum_{l=1}^{m} \frac{a_{il} a_{jl}}{k_l}, \tag{10}$$

where $k_j$ and $k_l$ denote the degree of object $j$ and user $i$, respectively.

To a target user, set the resources allocated to the collected objects and uncollected ones as 1 and 0, respectively. In this way, every user would have an $n$-dimensional vector, representing the personal information. Denote the personal vector by $\overrightarrow{f}$, then the resource is distributed in the following way:

$$\overrightarrow{f}' = \mathbf{W}\overrightarrow{f}. \tag{11}$$

Sort all of the uncollected objects according the predicted scores obtained in Eq. (11).

A benchmark data set, namely *MovieLens* (http://www.grouplens.org), which consists of 1682 movies (objects) and 943 users, is used to test the presented algorithm. The users vote movies by discrete ratings from 1 to 5. Here a coarse-graining method [56, 57] is applied: 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 $\geq 3$, that is, the user-object (user-movie) bipartite network after the coarse gaining contains 85250 edges. The data set is randomly divided into two parts: the training set contains 90% of the data, and the remaining 10% of data constitutes the probe. The average ranking score, $\langle k \rangle$, of the Global Rank Method is 0.136, the collaborative filtering algorithm based on the Pearson coefficient is 0.120, while the result obtained by the structure-based algorithm is 0.106, which indicates that the structure-based algorithm has the highest accuracy among those three algorithms.

**4.3. Effect of the object degree to the recommendation results.** In the last section, all the collected object would be allocated a unit resource 1. In other words, to every user, the recommendation abilities of all collected objects are equal. However, the number of collections of the objects are different. Some objects are popular, while the others are unpopular. *Zhou et al.* [57] enhanced the accuracy by eliminating the influences of the popular objects. To one target user $i$, set the initial resources of its collected objects as $f_j = a_{ji} k_j^{\beta}$, where $k_j$ is the degree of the object $j$ and $\beta$ is a tunable parameter. If $\beta > 0$, the influences of the popular objects would be enhanced; otherwise, their influences would be reduced. If $\beta = 0$, this method degenerates to the previous structure-based algorithms.

The numerical results on the *MovieLens* data showed that, when $\beta = -0.8$, the accuracy would dramatically increase to 0.0972, which increased about 8% compared with the algorithm in [56]. It should be emphasized that, when the predicted scores are equal, it is significant to recommend the objects with smaller degrees. Take the movies for example, in general, the users like the well-known films, but they also can get the film information from many other channels, such as the broadcast, TV and Internet. While for the unpopular films, it is hard for the users to find them without the recommender system's help. From this point of view, it is more meaningful to present the "hidden" information to the users. Back to the algorithm, it is better to recommend the objects with smaller degrees at the same accuracy level. In the real recommender systems, the recommendation list should not be longer than 100. For example, the personal music recommender system of *Yahoo* gives a list of 40 songs, while the smart social bookmark system only recommends 20 bookmarks to the users. Therefore, it makes more sense to evaluate the average object degree only at the top positions. To three special list lengths $L = 10, 20$ and 50, all the average object degrees increase with the parameter $\beta$ monotonously.

**4.4. Ultra accuracy algorithms by eliminating redundant correlations.** Take the movie for example, for simplicity, assume a user's appetite on a movie is determined by two factors: actor and director. In particular, assume a target user like the movies with actor $A$ or director $B$, and there are two movies. One has the actor $A$, denotes by $M1$, the other one has the director $B$, denotes by $M2$. If the movie $M3$ is acted by $A$ and directed by $B$, then both of the $M1$ and $M2$ would affect the movie $M3$, whose totally received recommendation power is 2. Assume another different scenario, the target user has watched two movies, both of which are directed by $B$ but acted by different actors. If the system recommends to the user one movie directed by $B$, the totally recommendation power would also be equal to 2. Obviously, there are redundant correlations in the latter case (the information obtained from the same director $B$). Although the recommended movies have the same power, the user would prefer the movie $M3$ since there is less redundant information. *Zhou et al.* [63] introduced a personal recommendation algorithm, namely the collaborative filtering, which has significantly higher accuracy than the classical algorithm. In this algorithm, the correlation resulting from a specific attribute may be repeatedly counted in the cumulative recommendations from different objects. By considering the higher-order correlations, they designed an effective algorithm that can, to some extent, eliminate the redundant correlations. The accuracy, measured by the ranking score, can be further improved by 23% in the optimal case.

Considering the second-order correlations, the resource allocation matrix can be expressed by [63]

$$\mathbf{W}^{'} = \mathbf{W} + \alpha \mathbf{W}^2, \tag{12}$$

where $\alpha$ is a tunable parameter. The final resource vector would be $\overrightarrow{f}^{\,'} = \mathbf{W}^{'}\overrightarrow{f}$. The numerical results indicate that, when $\alpha = -0.75$, the algorithm reaches its optimum and the corresponding $\langle r \rangle = 0.082$. The negative $\alpha$ supported the above analysis. A natural question arises: whether it is reasonable to consider the high-order correlations? Then they extended the algorithm to the following case

$$\mathbf{W}^{'} = \mathbf{W} + \alpha \mathbf{W}^2 + \beta \mathbf{W}^3, \tag{13}$$

where $\beta$ is also a parameter. Further numerical results showed that, in the optimal case, the algorithmic accuracy could be further enhanced by about 1%–2%. However, if we take the high-order correlations into account, much of the computational resources must be spent to achieve the exact results. Therefore, it is reasonable to only consider the second-order correlations in reality.

Based on the bipartite network, *Liu et al.* [64, 65] presented a modified collaborative filtering (MCF) algorithm, which has remarkably higher accuracy than the standard collaborative filtering. In MCF, instead of the standard Pearson coefficient, the user-user similarities, $s_{ij}$, are obtained by a diffusion process, which can be expressed by the following formulation.

$$s_{ij} = \frac{1}{k_j} \sum_{l=1}^{n} \frac{a_{li} a_{lj}}{k_l}, \tag{14}$$

where $k_j$ and $k_l$ denote the degrees of the objects and users. Furthermore, by considering the second-order similarities, they designed an effective algorithm that suppresses the influence of mainstream preferences. The similarity between two users is, in principle, an integration of many underlying similar tastes. For two arbitrary users, the very specific yet common tastes shall contribute more to the similarity measurement than those mainstream tastes. Assume that there are three users $A$, $B$, and $C$. All of them have selected object 1, and both of users $A$ and $C$ have selected the object 2. Both 1 and 2 contribute to the similarity between $A$ and $C$. Since 1 is the mainstream preference, it also contributes to the similarities between $A$ and $B$, as well as $B$ and $C$. Tracking the path $A \rightarrow B \rightarrow C$, the preference 1 also contributes to the second-order similarity between $A$ and $C$. Statistically speaking, two users sharing many mainstream preferences should have high second-order similarity, therefore the influence of mainstream preferences could be suppressed by taking account of the second-order similarity. Here, for simplicity, a linear form is used as:

$$\mathbf{H} = \mathbf{S} + \lambda \mathbf{S}^2, \tag{15}$$

where $\mathbf{H}$ is the new similarity matrix, $\mathbf{S} = \{s_{ij}\}$ is the first-order similarity defined as Eq. (15), and $\lambda$ is a tunable parameter. As discussed before, we expect the accuracy to be improved at some negative $\lambda$. The corresponding accuracy, measured by the ranking scores, is further improved by 24.9% in the optimal case.

## 5. Hybrid system

All the aforementioned algorithms used in reality have their own deficiencies. Actually, most real systems combine different algorithms together, named hybrid algorithms. The numerical simulations on some real data set indicate that the hybrid algorithms outperform the independent ones in accuracy. The most common forms of recommender systems are based on the collaborative filtering and content-based algorithms.

**5.1. Integrated with the independent systems.** One of the hybrid methods is to use them independently, and then combine the results into one. Some systems linearly combine the predicted scores into one [67, 68, 69], while others only present the optimal recommendation list, such the *Daily Learner system* [16] and the reference recommender system [17].

**5.2. Adding content-based algorithms into CFR.** Most of the hybrid systems, including Fab [43] system, are based on the content-based collaborative filtering algorithms. In these systems, the collaborative filtering algorithms are not based on the common collected objects, but rely on the user profiles [69], which could overcome the sparsity problem. Another advantage lies on the fact that, if the similarities between the object profiles and user profiles are very high, these systems would recommend them to the user directly even when these objects have never been marked before [43]. Based on the text mining algorithm, *Melville et al.* [70] added an additional score vector to every user. The users with high additional scores would be recommended with high priority. *Yoshii et al.* [71] recommended the music to the users by using the collaborative filtering algorithms and the audio analysis technology. By adding the domain ontology into the CFR, *Girardi* and *Marinho* [72] presented a Web recommender system. In addition, some systems added the content analysis technologies into the SBR [73, 74]. These websites obtained the content by synthesizing the Tag and keywords semantic information together.

**5.3. Other hybrid systems.** Based on the content-based and collaborative filtering algorithms, *Basu et al.* [66] constructed the user-movie bipartite correlations. The user array gathered the user information, whose interests are common with the target user, while the movie array collected the common properties of the movies. The correlations between users and movies are divided by like and dislike. In this way, the system predicted whether a user like the a movie or not. *Popescul et al.* [75] and *Schein et al.* [69] presented a probabilistic model for unified collaborative and content-based algorithms. This method expressed the user interests into lots of topics by using the latent sematic analysis, in this way, this method solved the sparsity problems efficiently. Based on the neural network, *Christakou et al.* [76] constructed a new recommender system. Some hybrid systems were constructed based on the knowledge-based methods [75, 77, 78]. In order to enhance the system accuracy, using the case-based reasoning method, *Burke* [77] recommended the menus and foods to the users. *Quickstep and Foxtrot system* [78] suggested the scientific papers to the readers according the the user ontological information. *Velasquez* and *Palade* [79] proposed the knowledge-based Web recommender system, where the Web content is abstracted first and the user browsing regulations are constructed based on the browsing behaviors, which is used to predict the interested content at the next step. Based on the text mining technologies, *Aciar et al.* [80] presented the knowledge and collaborative hybrid system. *Felfernig et al.* [4] presented an automatic answering system, named *CWAdvisor*. The system could find the user interests from dialogs, then the most correlative objects are recommended. *Mirzadeh* and *Ricci* [81] gave the personal recommendation by using the interactive management consulting. *Wang* and *Chang* [82] constructed a knowledge recommender system based on the visual research groups, which can recommend the explicit and implicit knowledge by using the content-based and collaborative filtering algorithms. The knowledge-based recommender systems mainly rely on the knowledge acquisition technology, which has hindered its development severely. However, with the development of XML and ontology technologies, it would be very promising.

## 6. Conclusion and discussions

Thanks to the rapid progression of the Internet and information technologies, we could easily find and use many resources, such as e-journals, e-mail, e-commerce, and so on. The Internet has changed the structure of the information world. Firstly, it is difficult for us to find the relevant or interesting information from the information world, while millions or billions pieces of information is hard to be found because of the lack of visitors. Such "hidden" information may be interesting to different users. Unfortunately, these users cannot find them without any assistance. The recommender system provides a bridge between such information and users. With the assistance of the recommender system, we can find the users' tastes and habits and then help them to find the objects they are interested in.

In this paper, four different recommendation algorithms are reviewed. Although most of them have been applied extensively, there are still many questions to be investigated from both the theoretical and application points of view. CFR faces the following three main challenges: cold-start, sparsity, and extendibility problems. CBR is constrained by the information acquisition technologies and overspecialization. SBR needs to solve the new users and objects problem.

Apart from all of the above problems, there are some common problems faced by all PRSs (see also the discussions in Ref. [83]). For example, both of the user and object information are dynamically changing. If all the information is saved statically, the system must re-compute the information when there is one new piece of information produced by the user or objects, which would cost a lot of system resources. One reasonable way to solve this problem is to present a dynamic algorithm based on local information updating. In addition, there exist many measurements to judge the recommendation results, such as Recall, Accuracy, and F-measure. For different systems, it is more important to select a compatible measurement to evaluate the system performance.

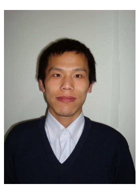
## References

[1] Maslov S and Zhang YC. Extracting hidden information from knowledge networks. Physical Review Letters, 2001, 87: 248701.
[2] Resnick P, Iacovou N, Suchak M, Bergstrom P, and Riedl J, GroupLens: An open architecture for collaborative filtering of netnews. Proc. ACM Conf. on Computer-Supported Cooperative Work, 1994, Chapel Hill, pp. 175–186.
[3] Hill W, Stead L, Rosenstein M, and Furnas G, Recommending and Evaluating Choices in a Virtual Community of Use. Proc. Conf. Human Factors in Computing Systems. Denver, 1995, pp. 194–201.
[4] Felfernig A, Friedrich G, Jannach D, and Zanker M, An integrated environment for the development of knowledge-based recommender applications. International Journal of Electronic Commerce, 2006, 11(2): 11–34.
[5] Good N, Schafer JB, Konstan JA, Borchers A, Sarwar B, Herlocker J, and Riedl J, Combining Collaborative Filtering with Personal Agents for Better Recommendations. Proc. Conf. Am. Assoc. Artificial Intelligence (AAAI-99), 1999, pp. 439–446.
[6] Rich E. User Modeling via Stereotypes. Cognitive Science, 1979, 3(4): 329–354.
[7] Goldberg D, Nichols D, Oki BM, and Terry D, Using Collaborative Filtering to Weave an Information Tapestry. Comm. ACM, 1992, 35(12): 61–70.
[8] Konstan JA, Miller BN, Maltz D, Herlocker JL, Gordon LR, and Riedl J, GroupLens: Applying Collaborative Filtering to Usenet News. Comm. ACM, 1997, 40(3) : 77–87.
[9] Shardanand U and Maes P, Social Information Filtering: Algorithms for Automating 'Word of Mouth'. Proc. Conf. Human Factors in Computing Systems. Denver, 1995: 210–217.
[10] Linden G, Smith B, and York J, Amazon.com Recommendations: Item-to-Item Collaborative Filtering. IEEE Internet Computing, 2003, 7(1): 76–80.
[11] Goldberg K, Roeder T, Gupta D, and Perkins C, Eigentaste: A Constant Time Collaborative Filtering Algorithm. Information Retrieval J., 2001, 4(2): 133–151.
[12] Terveen L, Hill W, Amento B, Mcdonald D, and Creter J, PHOAKS: A System for Sharing Recommendations. Comm. ACM, 1997, 40(3): 59–62.
[13] Breese JS, Heckerman D, and Kadie C, Empirical Analysis of Predictive Algorithms for Collaborative Filtering. Proc. 14th Conf. Uncertainty in Artificial Intelligence. Madison, 1998: 43–52.
[14] Delgado J and Ishii N, Memory-Based Weighted-Majority Prediction for Recommender Systems. Proc. ACM SIGIR '99 Workshop Recommender Systems: Algorithms and Evaluation, 1999.
[15] Nakamura A and Abe N, Collaborative Filtering Using Weighted Majority Prediction Algorithms. Proc. 15th Int'l Conf. Machine Learning, Madison, 1998: 395–403.
[16] Billsus D and Pazzani M, User Modeling for Adaptive News Access. User Modeling and User-Adapted Interaction, 2000, 10(2-3): 147–180.
[17] Getoor L and Sahami M, Using Probabilistic Relational Models for Collaborative Filtering. Proc. Workshop Web Usage Analysis and User Profiling, San Diego, 1999.
[18] Hofmann T, Collaborative Filtering via Gaussian Probabilistic Latent Semantic Analysis. Proc. 26th Ann. Int'l ACM SIGIR Conf., Toronto, 2003: 259–266.
[19] Marlin B, Modeling User Rating Profiles for Collaborative Filtering. Proc. 17th Ann. Conf. Neural Information Processing Systems, 2003.
[20] Pavlov D and Pennock D, A Maximum Entropy Approach to Collaborative Filtering in Dynamic, Sparse, High-Dimensional Domains. Proc. 16th Ann. Conf. Neural Information Processing Systems, 2002. (http://books.nips.cc/papers/files/nips15/AP06.pdf)
[21] Cohen WW, Schapire RE, and Singer Y, Learning to Order Things. J. Artificial Intelligence Research, 1999, 10: 243–270.

[22] Freund Y, Iyer R, Schapire RE, and Singer Y, An Efficient Boosting Algorithm for Combining Preferences. Journal of Machine Learning Research, 2003, 4: 933–969.
[23] Jin R, Si L, and Zhai C, Preference-Based Graphic Models for Collaborative Filtering. Proc. 19th Conf. Uncertainty in Artificial Intelligence (UAI 2003), Acapulco, 2003: 329–336.
[24] Jin R, Si L, Zhai C, and Callan J, Collaborative Filtering with Decoupled Models for Preferences and Ratings. Proc. 12th Int'l Conf. Information and Knowledge Management (CIKM 2003), New Orleans2003309–316.
[25] Sarwar B, Karypis G, Konstan J, and Riedl J, Item-Based Collaborative Filtering Recommendation Algorithms. Proc. 10th Int'l WWW Conf., Hong Kong, 2001: 1–5.
[26] Lee TQ, Park Y, and Park YT, A time-based approach to effective recommender systems using implicit feedback. Expert Systems with Applications, 2008, 34(4): 3055–3062.
[27] Chen YL and Cheng LC, A novel collaborative filtering approach for recommending ranked items. Expert Systems with Applications, 2008, 34(4): 2396–2405.
[28] Yang MH and Gu ZM, Personalized recommendation based on partial similarity of interests. Lecture Notes in Computer Science, 2006, 4093: 509–516.
[29] Aggarwal CC, Wolf JL, Wu KL, and Yu PS, Horting Hatches an Egg: A New Graph-Theoretic Approach to Collaborative Filtering. Proc. Fifth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining, San Diego, 1999: 201–212.
[30] Deshpande M and Karypis G, Item-Based Top-N Recommendation Algorithms. ACM Trans. Information Systems, 2004, 22(1): 143–177.
[31] Ungar LH and Foster DP, Clustering Methods for Collaborative Filtering. Proc. Recommender Systems, Papers from 1998 Workshop, Technical Report WS-98-08, Menlo Park, 1998: 84–88.
[32] Chien YH and George EI, A Bayesian Model for Collaborative Filtering. Proc. Seventh Int'l Workshop Artificial Intelligence and Statistics, 1999.
[33] Hofmann T, Latent Semantic Models for Collaborative filtering. ACM Trans. Information Systems, 2004, 22(1): 89–115.
[34] Si L and Jin R, Flexible Mixture Model for Collaborative filtering. Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003), Washington DC, 2003.
[35] Shani G, Brafman R, and Heckerman D, An MDP-Based Recommender System. The Journal of Machine Learning Research, 2005, 6: 1265–1295.
[36] Kumar R, Raghavan P, Rajagopalan S, and Tomkins A, Recommendation Systems: A Probabilistic Analysis. J. Computer and System Sciences, 2001, 63(1): 42–61.
[37] Yu K, Xu X, Tao J, Ester M, and Kriegel H, Instance Selection Techniques for Memory-Based Collaborative Filtering. Proc. Second SIAM Int'l Conf. Data Mining (SDM '02), 2002.
[38] Manouselis N and Costopoulou C, Experimental analysis of design choices in multiattribute utility collaborative filtering. International journal of pattern recognition and artificial intelligence, 2007, 21(2): 311–331.
[39] Chen YL, Cheng LC, and Chuang CN, A group recommendation system with consideration of interactions among group members. Expert systems with applications, 2008, 34(3): 2082–2090.
[40] Baeza-Yates R and Ribeiro-Neto B, Modern Information Retrieval. Addison-Wesley, Wesley Press, 1999.
[41] Salton G, Automatic Text Processing. Addison-Wesley, 1989.
[42] Belkin N and Croft B, Information Filtering and Information Retrieval. Comm. ACM, 1992, 35(12): 29–37.
[43] Balabanovic M and Shoham Y, Fab: Content-Based, Collaborative Recommendation. Comm. ACM, 1997, 40(3): 66–72.
[44] Pazzani M and Billsus D, Learning and Revising User Profiles: The Identification of Interesting Web Sites. Machine Learning, 1997, 27: 313–331.
[45] Mooney RJ, Bennett PN, and Roy L, Book Recommending Using Text Categorization with Extracted Information. Proc. Recommender Systems Papers from 1998 Workshop, Technical Report WS-98-08, 1998.
[46] Park HS, Yoo JO, and Cho SB, A Context-Aware Music Recommendation System Using Fuzzy Bayesian Networks with Utility Theory. Lecture Notes in Computer Science, 2006, 4223: 970–979.
[47] Somlo G and Howe A, Adaptive Lightweight Text Filtering. Proc. Lecture Notes in Computer Science, 2001, 2189: 319–329.
[48] Zhang Y, Callan J, and Minka T, Novelty and Redundancy Detection in Adaptive Filtering. Proc. 25th Ann. Int'l ACM SIGIR Conf., Tampere , 2002: 81–88.

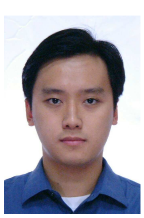
[49] Robertson S and Walker S, Threshold Setting in Adaptive Filtering. J. Documentation, 2000, 56: 312–331.
[50] Zhang Y and Callan J, Maximum Likelihood Estimation for Filtering Thresholds. Proc. 24th Ann. Int'l ACM SIGIR Conf., New Orleans, 2001: 294–302.
[51] Chang YI, Shen JH, and Chen TI, A data mining-based method for the incremental update of supporting personalized information filtering. Journal of Information Science and Engineering, 2008, 24(1): 129–142.
[52] Degemmis M, Lops P, and Semeraro G, A content-collaborative recommender that exploits WordNet-based user profiles for neighborhood formation. User Modeling and User-Adapted Interaction, 2007, 17(3): 217–255.
[53] Kazienko P and Adamski M, AdROSA - Adaptive personalization of web advertising. Information Sciences, 2007, 177(11): 2269–2295.
[54] Ricci F and Nguyen QN, Acquiring and revising preferences in a critique-based mobile recommender system. IEEE Intelligent systems, 2007, 22(3): 22–29.
[55] Martínez L, Pérez LG, and Barranco M, A multigranular linguistic content-based recommendation model: Research Articles. International Journal of Intelligent Systems, 2007, 22(5): 419–434.
[56] Zhou T, Ren J, Medo M, and Zhang YC, Bipartite network projection and personal recommendation. Phys. Rev. E 2007, 76: 046115.
[57] Zhou T, Jiang LL, Su RQ, and Zhang YC, Effect of initial configuration on network-based recommendation. Europhys. Lett., 2008, 81: 58004.
[58] Huang Z, Chen H, and Zeng D, Applying Associative Retrieval Techniques to Alleviate the Sparsity Problem in Collaborative Filtering, IEEE Trans.Inf.Syst., 2004, 22(1): 116–142.
[59] Huang Z, Zeng D, and Chen H, Analyzing consumer-product graphs: empirical findings and applications in recommender systems. Management science, 2007, 53(7): 1146–1164.
[60] Zhang YC, Blattner M, and Yu YK, Heat Conduction Process on Community Networks as a Recommendation Model. Phys. Rev. Lett., 2007, 99: 154301.
[61] Zhang YC, Medo M, Ren J, Zhou T, Li T, and Yang F, Recommendation model based on opinion diffusion. Europhys. Lett., 2007, 80: 68003.
[62] Zhou T, Liu JG, and Kuscsik Z, Being accurate is not enough: measuring and optimizing the diversity of recommendations. arXiv: 0808.2670.
[63] Zhou T, Su RQ, Liu RR, Jiang LL, Wang BH, and Zhang YC, Ultra accurate personal recommendation via eliminating redundant correlations. arXiv: 0805.4127.
[64] Liu JG, Zhou T, Wang BH, and Zhang YC, Highly accurate recommendation algorithm based on high-order similarities. arXiv: 0808.3726.
[65] Liu JG and Wang BH, A spreading activation approach for collaborative filtering. Int. J. Mod. Phys. C, 2009, 20(2): 285–293.
[66] Basu C, Hirsh H, and Cohen W, Recommendation as Classification: Using Social and Content-Based Information in Recommendation. Papers from 1998 Workshop, Technical Report WS-98-08, AAAI Press 1998: 714–720.
[67] Claypool M, Gokhale A, Miranda T, Murnikov P, Netes D, and Sartin M, Combining Content-Based and Collaborative Filters in an Online Newspaper. Proc. ACM SIGIR '99 Workshop Recommender Systems: Algorithms and Evaluation, Berkeley 1999.
[68] Pazzani M, A Framework for Collaborative, Content-Based, and Demographic Filtering. Artificial Intelligence Rev., 1999, 13(5-6): 393–408.
[69] Schein AI, Popescul A, Ungar LH, and Pennock DM, Methods and Metrics for Cold-Start Recommendations. Proc. 25th Ann. Int'l ACM SIGIR Conf., Tampere, 2002: 253–260.
[70] Melville P, Mooney RJ, and Nagarajan R, Content-Boosted Collaborative Filtering for Improved Recommendations. Proc. 18th Nat'l Conf. Artificial Intelligence, Edmonton, 2002: 187–192.
[71] Yoshii K, Goto M, Komatani K, Ogata T, and Okuno HG, An efficient hybrid music recommender system using an incrementally trainable probabilistic generative model. IEEE Transactions on Audio Speech and Language Processing, 2008, 16(2): 435–447.
[72] Girardi R and Marinho LB, A domain model of Web recommender systems based on usage mining and collaborative filtering. Requirements Engineering, 2007, 12(1): 23–40.
[73] Cattuto C, Loreto V, and Pietronero L, Semiotic dynamics and collaborative tagging. PNAS, 2007, 104(5): 1461–1464.
[74] Zhang ZK, Lü LY, Liu JG, and Zhou T, Empirical analysis on a keyword-based semantic system. Eur. Phys. J. B, 2008, 66: 557C-561.

[75] Popescul A, Ungar LH, Pennock DH, and Lawrence S, Probabilistic Models for Unified Collaborative and Content-Based Recommendation in Sparse-Data Environments. Proc. 17th Conf. Uncertainty in Artificial Intelligence, 2001: 437–444.
[76] Christakou C, Vrettos S, and Stafylopatis A, A hybrid movie recommender system based on neural networks. International Journal on Artificial Intelligence Tools, 2007, 16(5): 771–792.
[77] Burke R, Knowledge-Based Recommender Systems. In A. Kent (ed.): *Encyclopedia of Library and Information Systems*. Vol. 69, Supplement 32, 2000.
[78] Middleton SE, Shadbolt NR, and de Roure DC, Ontological User Profiling in Recommender Systems. ACM Trans. Information Systems, 2004, 22(1): 54–88.
[79] Velasquez JD and Palade V, Building a knowledge base for implementing a web-based computerized recommendation system. International Journal on Artificial Intelligence Tools, 2007, 16(5): 793–828.
[80] Aciar S, Zhang D, Simoff S, and Debenham J, Basing Recommendations on Consumer Product Reviews Informed Recommender: Basing Recommendations on Consumer Product Reviews. IEEE Intelligent systems, 2007, 22(3): 39–47.
[81] Mirzadeh N and Ricci F, Cooperative query rewriting for decision making support and recommender systems. Applied artificial intelligence, 2007, 21(10): 895–932.
[82] Wang HC and Chang YL, PKR: A personalized knowledge recommendation system for virtual research communities. Journal of Computer Information Systems, 2007, 48(1): 31–41.
[83] Adomavicius G and Tuzhilin A, Toward the next generation of recommander systems: A survey of the state-of-the-art and possible extensions. IEEE Trans. on Knowledge and Data Engineering, 2005, 17(6): 734–749.



**Dr. Jian-Guo Liu**, received his PhD degree in Management Science and Engineering from the Institute of Systems Engineering, Dalian University of Technology (DLUT) in 2007. He is currently a Postdoc at the Department of Modern Physics, University of Science and Technology of China. His current research interests include personal recommendation algorithm, complex network and knowledge management. He has published more than 50 refereed journal and conference papers in these areas, including 20 SCI index papers.

**Dr. Michael Z. Q. Chen**, was born in Shanghai. He graduated from Nanyang Technological University, Singapore, in 2003 with a B.Eng. degree in Electrical and Electronic Engineering, and from Cambridge University in 2007 with a Ph.D. degree in Control Engineering. He is currently a Lecturer in the Department of Engineering at the University of Leicester, England. He is a Fellow of the Cambridge Philosophical Society, a Life Fellow of the Cambridge Overseas Trust, and a member of the IEEE. Since 2008, he has been an Associate Editor of the IES Journal B–Intelligent Devices & Systems and a reviewer of the IEEE Transactions on Circuits & Systems, Automatica, International Journal of Adaptive Control & Signal Processing, and Journal of Sound & Vibration, amongst others. He is also a member of IEEE and an external reviewer for the Research Grants Council of Hong Kong. His research interests include: passive network synthesis, vehicle suspensions, complex networks, and statistical mechanics.

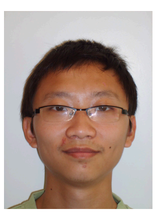


**Mr. Jianchi Chen**, received the B.S. degree in Electrical Engineering from Wuhan University in 2001. From 2001 to 2003, he worked as an engineer in the State Power Cooperation in China. He received his M.Phil. degree in Control Engineering from the University of Liverpool in 2004. He is currently pursuing a Ph.D. degree in the University of Leicester. His research interests include robust control, control of aerospace systems, convex optimization, and complex networks.

**Mr. Fei Deng**, is an associate professor of school of geodesy and geoinformatics, wuhan university. Research on computer vision and remote sensing.

**Dr. Hai-Tao Zhang**, was born in 1977. He graduated from University of Science and Technology of China in 2005 with a Ph.D. degree in Control Engineering. In 2004, he was a visiting scholar in IBM China Research Lab. In 2006 he was a Senior Research Assistant of MEEM Dept, City University of Hong Kong, working mostly on distributed model predictive control. In 2007, he was a Post-Doc Research Scholar at Department of Engineering and Wolfson College, University of Cambridge. From February till May he was a Research Fellow at City University of HongKong. From 2005 till now he is an Associate Professor of Control Engineering at the Huazhong University of Science and Technology, P.R. China. In 2004, he won President Prize of the Chinese Academy of Science. He is the associate editor of IST Transactions of Control Engineering-Theory and Applications, and a referee of IEEE Trans. on Automatic Control, IEEE Trans. on Automation Science and Engineering, IEEE CDC, IEEE ACC, etc.

**Mr. Zi-ke Zhang**, received the B.S. degree in Information System from Beijing Information Technology Institute in 2004. He received his M.S. degree in Management Science and Engineering from Renmin University of China in 2007. He is currently pursuing a Ph.D. degree in the University of Fribourg. His research interests include collaborative tagging systems, recommender systems and complex networks.



**Mr. Tao Zhou**, was born in Sichuan province 1983. He received his Bachelor degree, majoring in theoretical physics, from the University of Science and Technology of China in 2005. His main research interests include complex networks, statistical mechanics of information systems, collective dynamics, human dynamics and econophysics. He has published more than 90 SCI journal papers, including about 30 Physical Reviews. His publications have gotten more than 750 citations from Web of Science. He is also a standing referee of many international journals, including Phys. Rev. E, Europhys. Lett. Physica A, Phys. Lett. A, Int. J. Mod. Phys. B/C, Chin. Phys. Lett., J. Korean Phys. Soc., etc.

Department of Modern Physics and Nonlinear Science Center, University of Science and Technology of China, Hefei 230026, PRC
*E-mail*: liujg004@ustc.edu.cn(Jian-Guo Liu)

Department of Engineering, University of Leicester, Leicester LE1 7RH, U.K.
*E-mail*: michael.chen@cantab.net(Michael Z.Q. Chen) and jc252@le.ac.uk(Jianchi Chen)

School of Geodesy and Geomatics, Wuhan University, Wuhan 430079, PRC
*E-mail*: dengf-wh@163.com(F Deng)

Department of Control Science and Engineering, Huazhong University of Science and Technology; Department of Engineering, University of Cambridge.
*E-mail*: zht@mail.hust.edu.cn, hz254@cam.ac.uk(Hai-Tao Zhang)

Department of Physics, University of Fribourg, Fribourg 1700, Switzerland
*E-mail*: zike.zhang@unifr.ch(Zi-Ke Zhang)

Department of Modern Physics and Nonlinear Science Center, University of Science and Technology of China, Hefei 230026, P. R. China; Department of Physics, University of Fribourg, Fribourg 1700, Switzerland
*E-mail*: zhutou@ustc.edu(Tao Zhou)