Expert Systems With Applications 85 (2017) 270–278

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Timeliness in recommender systems

Fuguo Zhang[a], Qihua Liu[a], An Zeng[b,*]

[a] School of Information Technology, Jiangxi University of Finance and Economics, Nanchang, 330013, P.R. China
[b] School of Systems Science, Beijing Normal University, Beijing, 100875, P. R. China

ARTICLE INFO

Article history:
Received 8 January 2017
Revised 26 March 2017
Accepted 15 May 2017
Available online 19 May 2017

Keywords:
Recommender systems
Bipartite networks
Timeliness
Data division

ABSTRACT

Due to the high efficiency in finding the most relevant online products for users from the information ocean, recommender systems have now been applied to many commercial web sites. Meanwhile, many recommendation algorithms have been developed to improve the recommendation accuracy and diversity. However, whether the recommended items are timely or not in these algorithms has not yet been well understood. To investigate this problem, we consider a temporal data division which divides the links to probe set and training set strictly according to the time stamp on links. We find that the recommendation accuracy of many algorithms are much lower in temporal data division than in the random data division.With a timeliness metric, we find that the low accuracy is caused by the tendency of these algorithms to recommend out-of-date items, which cannot be detected with the random data division. To solve this problem, we improve the considered recommendation algorithms with a timeliness factor. The resulting algorithms can strongly suppress the probability of recommending obsolete items. Meanwhile, the recommendation accuracy is substantially enhanced.

© 2017 Elsevier Ltd. All rights reserved.

1. Introduction

The digital revolution brought to us what is known as "information overload", i.e. there is too much information for a single individual to deal with. As a result, nowadays there is hardly an e-commerce website without some form of information filtering and recommendation service (Xiao & Benbasat, 2007).Thanks to the Web 2.0 and Web applications, the recommender systems have been achieving rapid development. The recommender systems can help users find the useful items from the online information ocean. The e-commerce development has also greatly promoted the advantages of recommender systems, such as the Amazon.com and eBay.com. Accurate and efficient recommendation algorithms can help us analyze the potential consumption trends of users, and eventually provide an effective personalized recommendation service for them. The e-commerce development has also greatly promoted the advantages of recommender systems, such as the Amazon.com and eBay.com. Accurate and efficient recommendation algorithms can help us analyze the potential consumption trends of users, and eventually provide an effective personalized recommendation service for them. So far, there are many efficient recommendation algorithms that have been proposed, such as the collaborative filtering (Bobadilla, Ortega, Hernando, & Gutirrez, 2013; Jeong, Lee, & Cho, 2010; Konstan et al., 1997), diffusion-based (Zhang, Blattner, & Yu, 2008; Zhou et al., 2010; Zhou, Ren, Medo, & Zhang, 2007), content-based (Salter & Antonopoulos, 2006; Serrano-Guerrero, Herrera-Viedma, Olivas, Cerezo, & Romero, 2011), trust-aware (Martinez-Cruz, Porcel, J., Moreno, & Herrera-Viedma, 2015; Massa & Bhattacharjee, 2004), social impact (Deng, Huang, & Xu, 2014; Qian, Feng, Zhao, & Mei, 2014) and tag-aware (Feng & Wang, 2012; Huang, Yeh, Lin, & Wu, 2014) algorithms.

Time-aware recommender systems (TARS) have been received increasing attention in recent years (Campos, Díez, & Cantador, 2014). Most studies about time-aware recommender system focus on the idea that the attraction of items to users in online systems will decay with time (Ding & Li, 2005; Koren, 2009; Xiang et al., 2010), which means that the user's most recent ratings on a neighborhood of similar items show her current trend on such items.

However, an important issue, the timeliness of the recommended items, has been overlooked in most of the literature. Every item has its life cycle. Some are short, such as the items in a news website. Some are long. For example, some popular films in the past like "Titanic" are still chosen by some customers now. We assume that a user tend to select a new item or an item long being listed in the website but still popular now. Therefore, a well-performed recommendation algorithm should be able to include those items in the recommendation lists, and a metric should be

* Corresponding author.
E-mail addresses: zhangfuguo.redbird@gmail.com (F. Zhang), qh_liu@163.com (Q. Liu), anzeng@bnu.edu.cn (A. Zeng).

http://dx.doi.org/10.1016/j.eswa.2017.05.038
0957-4174/© 2017 Elsevier Ltd. All rights reserved.

designed to measure such timeliness feature of the recommendation algorithms. The previous recommendation metrics such as accuracy and diversity, however, fail to capture such timeliness feature.

In order to address the above problem, in this paper we propose a timeliness metric which measures whether the recommended items are timely. We examine the timeliness metric of many well-known recommendation algorithms with the temporal data division, i.e. the real data is divided to the training set and probe set strictly according to time. We find that most of the existing recommendation algorithms have very low timeliness value, including even the Heat Conduction algorithm which is supposed to recommend unpopular yet relevant items. We finally develop an effective framework to improve the timeliness of the existing recommendation algorithms. Interestingly, we find that the recommendation accuracy of the methods are simultaneously increased. The results are consistent when the methods are tested in different real data sets. The contributions of this study are as follows: (1) We study the performance of some representative recommendation algorithms with the temporal data division, and find that the recommendation accuracy with temporal data division is much lower than that with random data division. The reason behind this is that the probe set in the temporal probe set has many new items, but traditional recommendation algorithms do not consider the timeliness of the recommend items. Therefore, the accuracy of most existing recommendation algorithms is low. (2) We proposed a timeliness metric for recommender systems, which measures whether the recommended items are timely, due to the fact that previous recommendation metrics such as accuracy and diversity fail to capture such timeliness feature. (3) To enhance the recommendation accuracy in the temporal data division, we propose a simple timeliness-based recommendation framework which can be applicable to any recommendation algorithm including CF, diffusion-based algorithms, content-based algorithms and hybrid algorithms. It directly modifies the recommendation score of items by incorporating their timeliness factor. We find that with this modification, the recommendation accuracy is largely improved.

The paper is organized as follows. In Section 2, we will give a brief review of the relevant works. In Section 3, we will introduce the timeliness metric and describe our timeliness based recommendation algorithms we are going to used. In Section 4, we will present the simulation results and discussion. Finally, we will conclude this work with a brief outlook of the future work in Section 5.

2. Related works

2.1. Recommendation algorithms

In the literature, there are many effective recommendation algorithms. Collaborative filtering (CF) is the most widely applied recommendation technology in real online systems (Bobadilla, Ortega, Hernando, & Gutirrez, 2013; Konstan et al., 1997; Ładyżyński & Grzegorzewski, 2015). In addition, the content-based (Salter & Antonopoulos, 2006; Serrano-Guerrero, Herrera-Viedma, Olivas, Cerezo, & Romero, 2011), trust-aware (Martinez-Cruz, Porcel, J., Moreno, & Herrera-Viedma, 2015; Massa & Bhattacharjee, 2004), social impact (Deng, Huang, & Xu, 2014; Qian, Feng, Zhao, & Mei, 2014) and tag-aware (Feng & Wang, 2012; Huang, Yeh, Lin, & Wu, 2014) are also frequently used recommendation technologies. Recently, the fruitful achievements of complexity theory, especially some physical methods such as mass diffusion (Lü & Liu, 2011; Shang, Lu, Zhang, & Zhou, 2009; Zeng, Vidmer, Medo, & Zhang, 2014; Zhou et al., 2010; Zhou, Ren, Medo, & Zhang, 2007) and heat conduction (Liu, Zhou, & Guo, 2011; Zhang, Blattner, & Yu, 2008), have attracted increasing attention from both computer science and physics community.

Researchers use the user-item bipartite network to model the online commercial system (Lü et al., 2012). The mass diffusion algorithm is a spreading process on the bipartite network, which has high accuracy but low personality and surprisal (Lü & Liu, 2011; Shang, Lu, Zhang, & Zhou, 2009; Zeng, Vidmer, Medo, & Zhang, 2014; Zhou et al., 2010; Zhou, Ren, Medo, & Zhang, 2007). The heat conduction method, another spreading process on bipartite network, has low accurate but high personality and surprisal (Liu, Zhou, & Guo, 2011; Zhang, Blattner, & Yu, 2008). In ref. (Zhou et al., 2010), the authors proposed a hybrid method to combine the mass diffusion and heat conduction which solve the apparent diversity-accuracy dilemma of recommender systems. In other words, a recommender system should not only consider to recommend the popular objects, but also the niche objects. After ref. (Zhou et al., 2010), many different methods have been proposed to achieve even better recommendation performance. For example, the preferential diffusion (Lü & Liu, 2011) and the biased heat conduction (Liu, Zhou, & Guo, 2011) have been designed to yield higher accuracy and larger diversity compare to the method in (Zhou et al., 2010). Moreover, the network manipulation has been shown to effectively solve the cold-start problem in recommendation (Zhang & Zeng, 2012). To enhance the efficiency of the recommendation process, the method to extract the information backbone (minimum structure) from online system is also designed (Zhang, Zeng, & Shang, 2013). Very recently, the long-term influence of the recommendation methods on the user-item bipartite network evolution is studied (Zeng, Chi, Medo, & Zhang, 2015). It is found that many personalized recommendation methods have reinforce effect on item degree distribution.

2.2. Evaluation metrics

As it is very costly to directly validate the effectiveness of the recommendation algorithms in online web sites, researchers have proposed the training-probe set data division framework. So far, most of the recommendation algorithms are examined with the random data division, i.e. the real data is randomly divided to a training set and a testing set (Herlocker, Konstan, Terveen, & Riedl, 2004). In a recent review (Yu, Zeng, Gillard, & Medo, 2015), it is mentioned that recommendation should be done with the data divided into the training set and probe set based on the time stamps on links. Focusing on the over-fitting problems for recommendation algorithms, Zeng et al. proposed a triple data division model in which the real data is divided into a training set, a learning set and a probe set (Zeng, Vidmer, Medo, & Zhang, 2014). The basic idea is to estimate users' parameters with the learning set and then applied the learned parameters to actually predict users' future objects in the probe set.

Many metrics have been proposed to measure the performance of the recommendation methods (Gunawardana & Shani, 2008). One main aspect to measure is the recommendation accuracy which is simple the overlap of the predicted items and the true future items selected by the users (Bobadilla, Ortega, Hernando, & Gutirrez, 2013). The effectiveness of most existing recommendation algorithms are judged by their accuracy (Herlocker, Konstan, Terveen, & Riedl, 2004). The recommendation diversity is another important aspect of the recommendation performance (Bradley & Smyth, 2001). It measures whether the recommended items are different from one user to another (which is usually called personality) and whether the recommended items are of small popularity (which is usually called surprisal). The recommendation diversity is now gradually accepted also by the computer scientists as a significant aspect for recommendation performance.

2.3. Time-aware recommendation algorithms

Since the purchase behaviors which happened long time ago could not truly reflect the current interests of the target user, many methods have been proposed to improve the recommendation accuracy with the time information. The first recommendation algorithm considering time information is the one proposed by Zimdars et al. who mapped the recommendation problem as a time series prediction problem (Zimdars, Chickering, & Meek, 2001). After that, most of the related work focused on the collaborative filtering (CF) method. The ways to improve CF are mainly heuristic-based (or memory-based) approaches (Adomavicius, Sankaranarayanan, Sen, & Tuzhilin, 2005) and model-based approaches (Koren, 2009). Recently, Song et al. investigated the impact of the time window on the training set on recommender algorithms. The experimental results indicate that by only selecting a portion of recent rating records as the training set, the accuracy could be improved a lot, and the diversity could be improved slightly (Song, Qiang, & Liu, 2015). For a recent review on the time-aware recommendation algorithms, see Ref. (Campos, Díez, & Cantador, 2014). Even though these methods can effective improve the recommendation accuracy, whether these existing algorithms can avoid recommending obsolete items still remains unknown. This is partially because a metric measuring the timeliness of the recommended items is missing. In addition, how to incorporate the time information in the diffusion-based algorithm is not well investigated. The contribution of this paper is to solve these two problems.

3. Timeliness metric and timeliness-based recommendation algorithms

3.1. Definition of timeliness metric

Every item has its product life cycle. If an item has passed its maturity and enters recession, it means that fewer and fewer customers will pay attention to it. A well-performed recommendation algorithm should be able to include those items not in recession. The existing recommendation algorithms treat items in recession the same as other items in the recommendation process. Here, we propose a new metric called timeliness to measure the extent to which the items are new or long listed in the market. For a item α, its timeliness T_α can be computed as

$$T_\alpha = \frac{1}{k_\alpha}\sum_{i\in U_\alpha}(t_{i\alpha} - t_0), \tag{1}$$

where k_α is the degree of item α representing the number of users selecting it, U_α is the set of users who selected α, $t_{i\alpha}$ is the time when i selects item α, t_0 is the starting time of the data set. Accordingly to this definition, if an item appeared long time ago but recently selected by many users, its timeliness can still be high. With this definition, one can estimate the timeliness of the items in a user i's recommendation list, as a metric evaluating the recommendation results. Mathematically, it reads

$$T_i(L) = \frac{1}{L}\sum_{\alpha\in O^i} T_\alpha, \tag{2}$$

where T_α is the timeliness of item α. O_i represents the recommendation list for user i, and L is the length of the recommendation list. The timeliness of the recommendation algorithm $T(L)$ is obtained by averaging $T_i(L)$ of each user. A high $T(L)$ indicates the strong timeliness of the recommended items. The reason of using the timeliness metric in recommendation is illustrated in Fig. 1. Neither the old relevant items in recession nor the new irrelevant items are good recommendations. The relevant and timely items are more likely to be liked by the users.

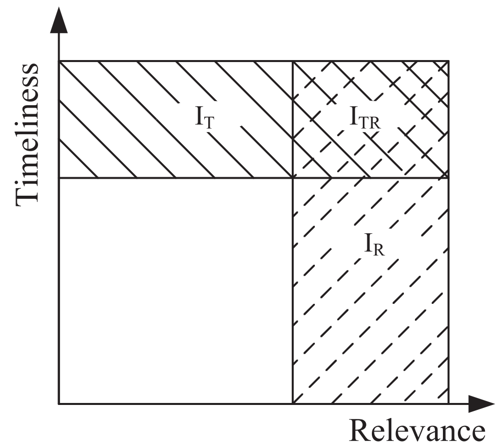

Fig. 1. The illustration of the reason of using the timeliness metric in recommendation.

3.2. Timeliness-based recommendation algorithms

As mentioned in Section 3.1,timeliness is a quantity measuring whether an item is still relevant in current time. A low timeliness value indicates that the item is already out-of-date. We assume that a user tend to select a timely item, which is not considered in traditional recommendation algorithm. Therefore,in this paper we propose a simple timeliness-based recommendation framework which can be applied to different existing recommendation algorithms.

Usually, each recommendation algorithm will obtain a recommendation score of an item α to user i, denoted as $f_{i\alpha}$. As in most algorithms, a value $f_{i\alpha}$ is calculated as the recommendation score of an item α to a user i. The basic idea of our approach is that the score of the out-of-date items should be suppressed. Therefore, we propose a straightforward modification of the score, which is simply the product of the $f_{i\alpha}$ and T_α (the timeliness of item α). As $f_{i\alpha}$ is a general score which needs to be calculated in any recommendation algorithm, our medication is actually applicable to any recommendation algorithm including CF, diffusion-based algorithms, content-based algorithms and hybrid algorithms. The timeliness-based recommendation approach simply modifies this score by

$$\tilde{f}_{i\alpha} = f_{i\alpha} \times T_\alpha \tag{3}$$

where T_α is the timeliness of item α. The recommendation list for user i can be obtained by sorting vector $\tilde{\mathbf{f}}_i$ in descending order. The timeliness-based recommendation approach can be applied to any recommendation algorithm (see the description of some representative recommendation algorithms below).

4. Numerical experiments

4.1. Recommendation algorithms

Among the existing recommendation algorithms, we consider five representative ones including the popularity-based recommendation (PR) method (Lü et al., 2012), the item-based collaborative filtering, the Mass diffusion method (Zhou, Ren, Medo, & Zhang, 2007), the heat conduction method (Zhang, Blattner, & Yu, 2008) and the hybrid method (Zhou et al., 2010) in this paper. Before introducing the recommendation algorithms, we first describe the notations for the user-item bipartite networks. This kind of network consists two types of nodes, i.e. user nodes and item nodes. If a user collects an item, a link is drawn between them. We consider a system of N users and M items represented by a bipartite network with adjacency matrix A, where the element $a_{i\alpha} = 1$, if a user i has collected an object α, and $a_{\alpha}=0$, otherwise (throughout this paper we use Greek and Latin letters, respectively, for item- and user-related indices).

The most straightforward method is the popularity-based recommendation (PR) method. In this method, each user is recommended with the top-L most popular items where L is the length of the recommendation list. The recommendation score of an item α to user i is simply $f_{i\alpha}=k_{\alpha}$ where k_{α} is the cumulative degree (popularity) of item α.

The collaborative filtering (CF) is a personalized recommendation algorithm which means that users are presented with different recommendation results. It consists of the user-based and item-based versions. In general, the item-based collaborative filtering (ICF) has higher accuracy. Therefore, in this paper we consider this algorithm. In ICF, the recommendation score of an unselected item is evaluated based on its similarity with the collected items of the target user. Here we define the similarity with the Cosine index (Salton & Mcgill, 1983) in the bipartite networks. The final recommendation score for each item can be written as

$$f_{i\alpha}=\sum_{\beta=1}^{M}s_{\alpha\beta}a_{i\beta}, \tag{4}$$

where $s_{\alpha\beta}$ is the cosine similarity between item α and β. The recommendation list for each user i is obtained by sorting $f_{i\alpha}$ in descending order.

The Mass diffusion and Heat conduction methods are both based on a two-step spreading process on the user-item bipartite network. The components of the adjacency matrix as $a_{i\alpha}$ and the vector with initial resources as $\mathbf{g}_i$ where component $g_{i\alpha}$ is the resource assigned to item α. When computing recommendation for user i, the resource vector is initialized as $g_{i\alpha}=a_{i\alpha}$, i.e., one unit of resource is assigned to each item collected by user i. The recommendation scores $\mathbf{f}_i$ are obtained as $\mathbf{f}_i=\mathbf{W}\mathbf{g}_i$. The difference between the Mass diffusion and Heat conduction methods lays on the matrix $\mathbf{W}$. In general, $\mathbf{W}$ can be written as

$$W_{\alpha\beta}=\frac{1}{k_{\alpha}^{1-\lambda}k_{\beta}^{\lambda}}\sum_{j=1}^{N}\frac{a_{j\alpha}a_{j\beta}}{k_j}, \tag{5}$$

where k_{β} is the degree of item β and k_j is the degree of user j. $\lambda\in[0,1]$ is a tunable parameter. As λ increases from 0 to 1, this so-called hybrid algorithm changes gradually from Heat conduction method (i.e. $\lambda=0$) to Mass diffusion method (i.e. $\lambda=1$).

4.2. Data

In this paper, we use data sets from two real online web sites. The MovieLens[1] data is about ratings of online users on movies. The level of rating from 1 to 5 as worst to best. We remove the ratings lower than 3 (Lü & Liu, 2011) (The rating lower than 3 means the user don't like the item). After filtering, the data contains 864,581 user-object pairs including 5000 users and 7533 item. The data has time information, with 1096 days in total. The Netflix data[2] is a random sampling of the whole records of user interaction in the Netflix website. It has the ratings of 4960 online users on 16,569 movies. We also carry out the filtering process by considering the link with ratings equal or above 3. After link filtering, there are 1,249,058 links left. This data also has time information, with 2183 days in total. To model the recommendation process, the real data is divided into two parts according to the time stamp: the earliest 90% links form the training set (ET) and the rest of the data (i.e. the latest 10% links) form the probe set (EP). The training set is treated as known information and the recommendation algorithms will be applied to it. The probe set will be used to evaluation the performance of the recommendation algorithms. Some main evaluation metrics are described below.

4.3. Metrics

The ranking score (RS) is one of the representative accuracy metrics (Zhou et al., 2010). For a target user i, the position for each of his link (i.e. his selected object) in the probe set is measured. Assume the rank of object α in i's recommendation list is $r_{i\alpha}$ and the total number of unselected objects is n_i, then the ranking score of this probe set link $i\alpha$ is $r_{i\alpha}/n_i$. The ranking score of the recommendation algorithm is obtained by

$$RS=\frac{1}{|E_P|}\sum_{i\alpha\in E_P}RS_{i\alpha}. \tag{6}$$

Clearly, a well-performed recommendation algorithm will place the probe set link in the top of users' recommendation lists. Therefore, the smaller RS, the more accurate the recommendation algorithm.

RS calculates the accuracy of the recommendation algorithms based on the whole ranking lists of the objects. However, in reality the recommender systems only show each user a short list of the most relevant objects. Therefore, whether the top ranking list fits users' interest is a more practical question. The precision (P) metric aims to evaluate the recommendation algorithms' accuracy in this aspect. For each user i, the precision of recommendation is calculated as

$$P_i(L)=\frac{d_i(L)}{L}, \tag{7}$$

where $d_i(L)$ represents the number of user i's probe set links in the top-L places in the recommendation list. The precision $P(L)$ of the recommendation algorithms can be obtained by averaging the individual precisions over all users with at least one link in the probe set. The higher the $P(L)$, the better the recommendation results.

Besides accuracy, diversity is also very important for recommendation algorithms. In this paper, we consider two standard diversity metrics: personalization and surprisal (Zhou et al., 2010). The personalization measures how users' recommendation lists are different from each other. Such difference is computed with the hamming distance. For user i and j, their hamming distance can be expressed as

$$D_{ij}(L)=1-\frac{C_{ij}(L)}{L}, \tag{8}$$

where $C_{ij}(L)$ is the number of common items in the top-L place of the recommendation list of i and j. According to the definition, $D_{ij}(L)$ is between 0 and 1, which are respectively corresponding to the cases where i and j have the same or an entirely different recommendation list. By averaging $D_{ij}(L)$ over all pairs of users, we obtain the personalization $D(L)$ of the recommendation algorithms. The higher $D(L)$, the more personalized the recommendations.

Surprisal measures the average popularity of the items in the recommendation list. This is based on the fact that users may have already known the popular objects from other channels. Therefore, a well-performed recommendation algorithm should help users discover unpopular objects. Therefore, the surprisal metric can be written as

$$I_i(L)=\frac{1}{L}\sum_{\alpha\in O^i}k_{\alpha}, \tag{9}$$

where O_i represents the recommendation list for user i, k_{α} represents the degree of the item α (i.e. number of links connecting to item α). The surprisal of the recommendation algorithm is obtained by averaging $I_i(L)$ of each user. A low $I(L)$ indicates recommendation of unpopular objects.

[1] http://www.grouplens.org/.

[2] https://www.netflix.com/.

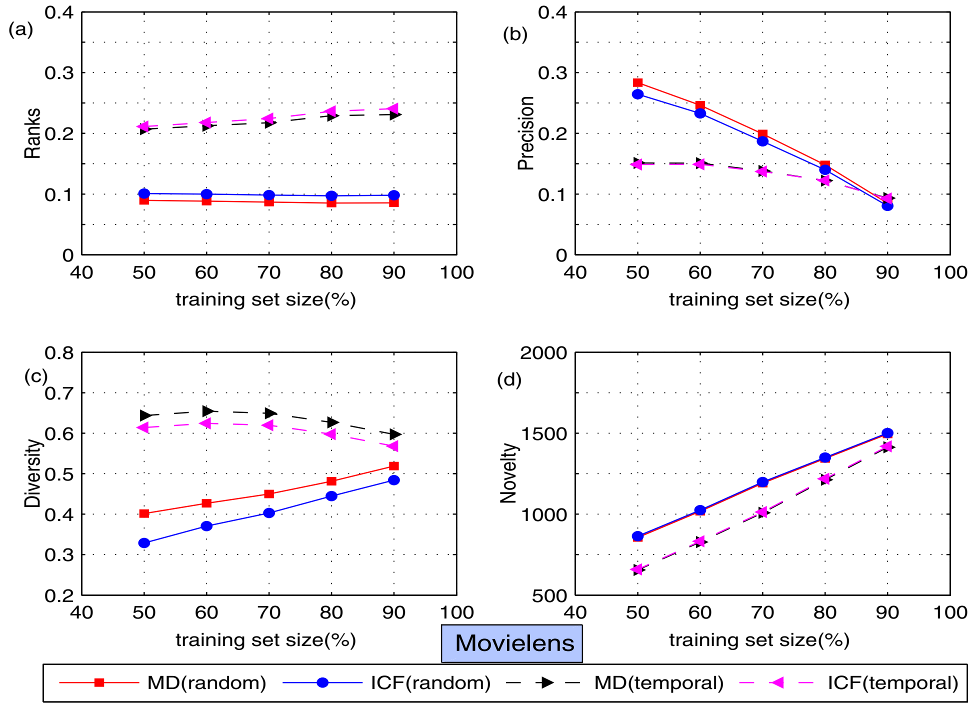

Fig. 2. (Color online) The recommendation performance ((a) ranking score; (b) precision; (c) personalization; (d) surprisal) of the mass diffusion and item-based collaborative filtering algorithms under the random data division and temporal data division. The network used in this figure is Movielens.

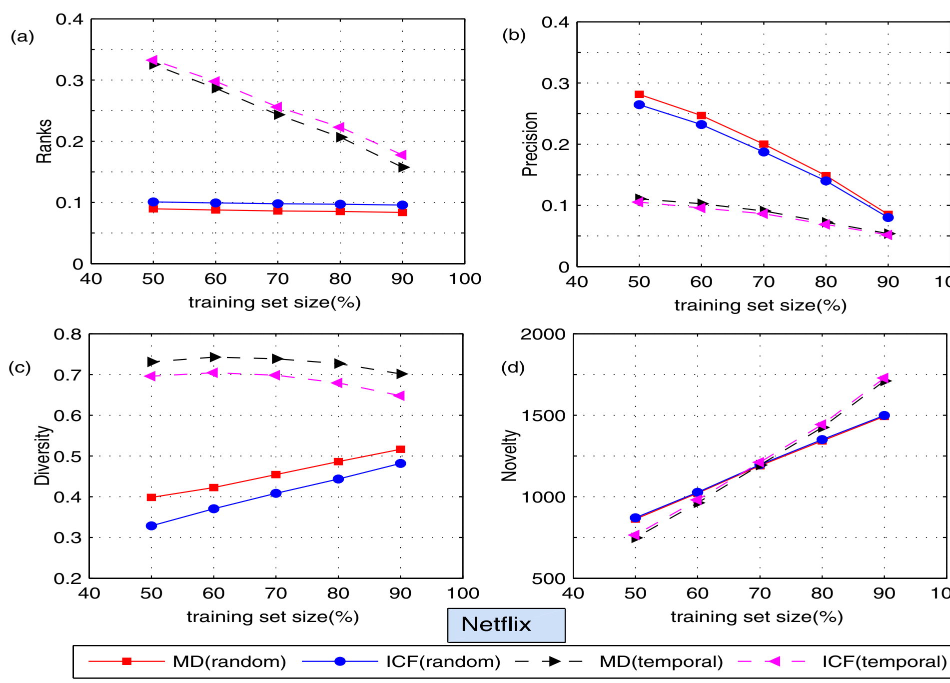

Fig. 3. (Color online) The recommendation performance ((a) ranking score; (b) precision; (c) personalization; (d) surprisal) of the mass diffusion and item-based collaborative filtering algorithms under the random data division and temporal data division. The network used in this figure is Netflix.

5. Results and discussion

To begin our analysis, we compare the recommendation results of the recommendation algorithms under the random data division and temporal data division, as shown in Figs. 2 and 3. As mentioned above, the temporal data division is based on the time stamps on links (i.e. earlier links are put in the training set and later links are put in the probe set). The random data division adjusts the size of the training set by directly controlling the number of links placed in the training set. The temporal data division, however, adjusts the size of the training set by selecting different testing time. For a fair comparison, we make sure that when the fraction of links in the training set is p in the random data division, the fraction of links in the training set is also p in the temporal data division. We use the Mass diffusion method and the item collaborative filtering method as examples, and show their performance with both kind of data divisions. In both Figs. 2 and 3, one can see that the recommendation performance is very different when a

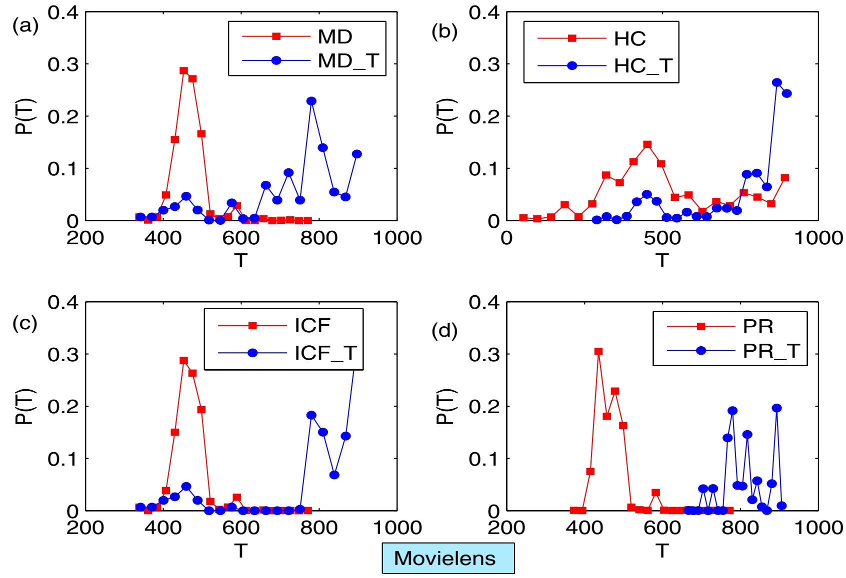

Fig. 4. (Color online) The distribution of T_α in users' recommendation lists when different recommendation algorithms are applied. The method marked with "T" is the timeliness-based version of the method. The network data used in this figure is Movielens.

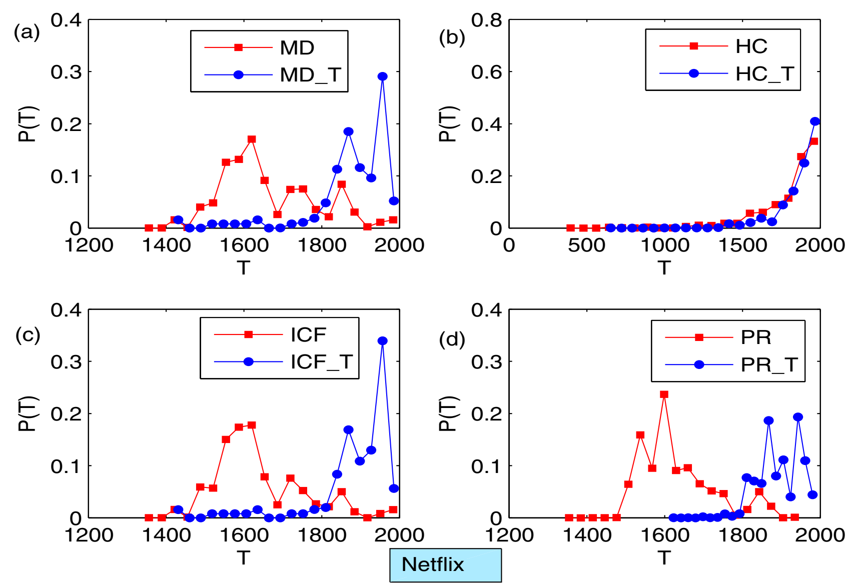

Fig. 5. (Color online) The distribution of T_α in users' recommendation lists when different recommendation algorithms are applied. The method marked with "T" is the timeliness-based version of the method. The network data used in this figure is Netflix.

recommendation algorithm is applied to the random data division and the temporal data division. Specifically, the recommendation accuracy is much lower in the temporal data division, indicating that the random data division cannot capture the recommendation process in real systems and real future items of online users are hard to predict.

We investigate the existing recommendation methods with respect to the timeliness of the recommended items. The Mass diffusion, Heat conduction, Collaborative filtering and Popularity-based methods are examined. For each method, we compute $T_\alpha = t_\alpha - t_0$ for each recommended item α in users' recommendation lists. We then compare the distribution of T_α obtained by different methods in Fig. 4 and Fig. 5. One can see that the distribution of T_α obtained by the Mass diffusion, Collaborative filtering and Popularity-based methods all have very small probability for large T, indicating that many recommended items by these methods are out-of-date. On the contrary, the distribution of T_α obtained by Heat conduction has a higher probability for large T, indicating that this method can recommend many timely items to users.

Based on the timeliness metric, we further design for each method a timeliness-based version that can include more timely items in the recommendation lists. As discussed in the method section, the recommendation algorithm will compute for each user i the recommendation scores of his/her unselected items α as $f_{i\alpha}$. The timeliness-based version of the recommendation algorithm is simply modifying the recommendation scores as $f_{i\alpha}(t_\alpha - t_0)$, and the recommendation list for user i is finally generated by sorting $f_{i\alpha} \times (t_\alpha - t_0)$ in descending order. Even though out-of-date items have high $f_{i\alpha}$, its $(t_\alpha - t_0)$ is very small. We show the timeliness distribution of the timeliness-based mass diffusion, timeliness-based heat conduction, timeliness-based item-based collaborative filtering, timeliness-based popularity-based algorithms in Figs. 4 and 5. One can clearly see that the timeliness-

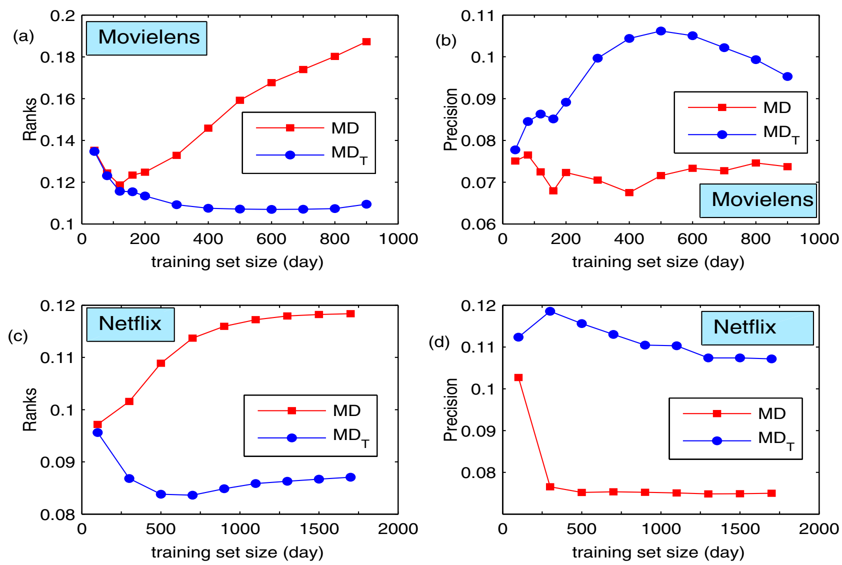

Fig. 6. (Color online) The dependence of ranking score and precision on the training set size when the traditional and timeliness-based methods are applied. The size of the training set is measured by the number of days' data in the training set. The probe set consists of 10% (this ratio is with respect to the total number of links in the data set) future links after the testing time. (a)(b) are the results of the Movielens data set. (c)(d) are the results of the Netflix data set.

based recommendation methods can provide much more timely items compared to their traditional counterparts.

We then further study whether the timeliness-based recommendation algorithms can improve the recommendation accuracy. To this end, we plot in Fig. 6 the dependence of ranking score and precision on the training set size (measured by the number of days' data in the training set) when the traditional and timeliness-based methods are applied. We consider Mass diffusion as an example in Fig. 6. One can see that the traditional and timeliness-based methods perform similarly when the training set is small. However, when the training set size is larger than a certain value, the performance of the traditional and timeliness-based methods start to split, i.e. the ranking score of the traditional method gradually becomes worse while the ranking score of the timeliness-based method decreases quickly to a small value and then becomes stable. The improvement in the recommendation accuracy is due to the fact that more timely items are presented in the recommendation lists and users tend to selected these items. The phenomenon is roughly similar when the accuracy is measured by Precision. In Fig. 6, we also compare the results in both Movielens and Netflix data sets, and the results are consistent.

Finally, we take into account the well-known Hybrid recommendation algorithm which is a combination of the Mass diffusion and Heat conduction methods with a tunable parameter λ (see the method section). We introduce another parameter controlling the effect of the timeliness factor in the corresponding timeliness-based method. Specifically, the recommendation score of items can be written as $f_{i\alpha} \times (t_\alpha - t_0)^\theta$. When $\theta = 0$, the timeliness-based method reduces to the traditional method. The larger θ is, the more weight are given to the timeliness of the items when the algorithms decides whether an item should be recommended. In Fig. 7(a) and (b), we first fix $\theta = 1$, and study the dependence of ranking score on the training set size when traditional Hybrid method and the timeliness-based Hybrid method are used. In both methods, λ is set to be the optimal value with respect to the ranking score. One can see that the behavior of the Hybrid method is similar to the Mass diffusion method, i.e. the timeliness-based Hybrid method remarkably outperforms the traditional Hybrid method in recommendation accuracy.

In Fig. 7(c) and (d), we study the effect of θ on the ranking score. Clearly, the ranking score achieves a minimum value at certain θ, meaning that to achieve the optimal recommendation accuracy one has to take into account both relevance (i.e. the diffusion scores obtained via the Hybrid algorithm) and timeliness of the items. In Fig. 7(e) and (f), we study the dependence of the ranking score on parameter λ. One can see that the optimal λ is very different in the random data division and the temporal data division. The optimal λ is closer to 0 in the traditional Hybrid method, indicating that the Mass diffusion method should be given more weight and the recommendation should be more biased to the unpopular items. On the contrary, the optimal λ is closer to 1 in the timeliness-based Hybrid method, indicating that the Heat conduction method plays a more important role and the recommendation should give priority to the popular items. The difference of λ is due to the fact that in the temporal data division, many items in the probe set are new items with small degree. Therefore, a more accurate recommendation should include more unpopular items in the recommendation list. The timeliness-based Hybrid method has already taken into account the timeliness of the items, so the recommended items are already new and the parameter λ doesn't have to be small (close to Heat conduction). In Table 1, we quantitatively compare the traditional and timeliness-based recommendation algorithms in Movielens and Netflix data sets. Clearly, all the five timeliness-based recommendation methods result in higher timeliness compared to their traditional counterparts. Meanwhile, all the five timeliness-based recommendation algorithms except Heat conduction method outperform the traditional recommendation algorithms on rank score metric.

6. Conclusion

In this paper, we study the performance of some representative recommendation algorithms with the temporal data division. After a testing time is set, the links appearing before this time consists of the training set and the links appearing after this time are placed in the probe set. We find that the recommendation performance of these algorithms is significantly different in the temporal data division and the traditional random data division. In general, the recommendation accuracy with temporal data division is much lower than that with random data division. The reason behind this is that the probe set in the temporal probe set has many new items. But traditional recommendation algorithms don't consider

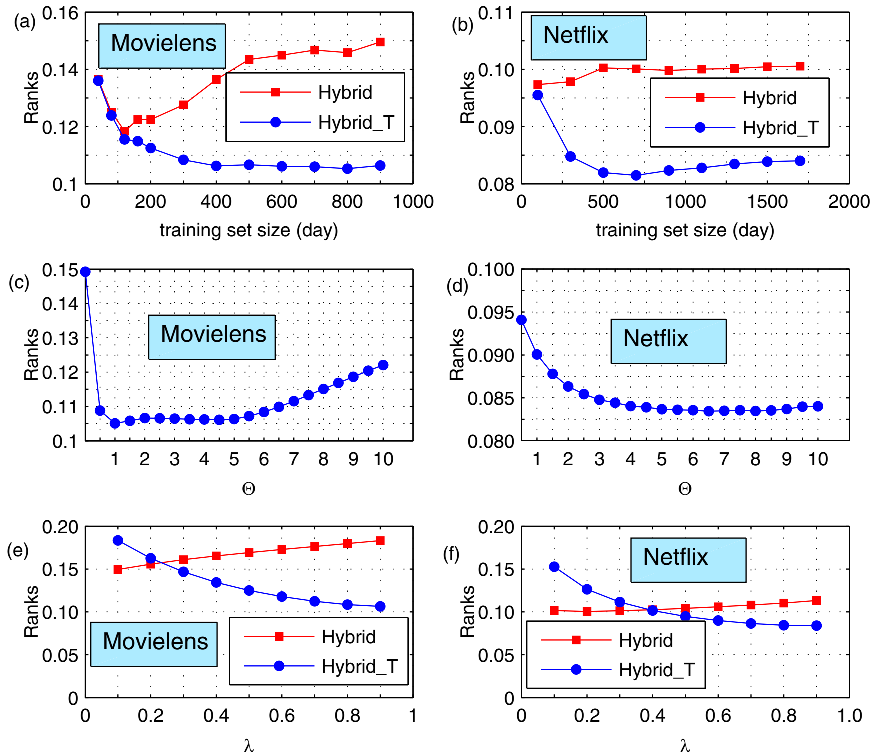

Fig. 7. (Color online) (a)(b) show the dependence of ranking score on the training set size when traditional Hybrid method and the timeliness-based Hybrid method are used in Movielens and Netflix, respectively. In these two figures, the parameter θ and λ are respectively set as 1 and the optimal value for ranking score. (c)(d) show the effect of θ on the ranking score in Movielens and Neflix. In these two figures, the parameter λ is set as the optimal value for ranking score. (e)(f) show the dependence of the ranking score on parameter λ in Movielens and Neflix. In these two figures, the parameter θ is set to be 1.

Table 1
Comparison of the traditional and timeliness-based recommendation algorithms in Movielens and Netflix data sets. For the *ranks* metrics, the smaller the better, while, regarding the *timeliness* metric, the higher the better. The algorithm with better performance is highlighted in bold font.

Method	Movielens (*ranks*)		Movielens (*timeliness*)		Netflix (*ranks*)		Netflix (*timeliness*)	
	Traditional	Timeliness-based	Traditional	Timeliness-based	Traditional	Timeliness-based	Traditional	Timeliness-based
PR	0.1929	**0.1188**	613.7	**810.0**	0.1440	**0.1049**	1631	**1908**
ICF	0.1927	**0.1212**	612.6	**747.9**	0.1386	**0.1005**	1640	**1888**
MD	0.1677	**0.1069**	614.5	**717.2**	0.1184	**0.0871**	1665	**1878**
HC	0.1634	**0.1307**	625.3	**729.1**	**0.1247**	0.1717	1807	**1919**
Hybrid	0.1450	**0.1061**	615.2	**665.0**	0.1006	**0.0840**	1671	**1864**

the timeliness of the recommend items. Therefore, the accuracy of most existing recommendation algorithms is low. To enhance the recommendation accuracy in the temporal data division, we propose a temporal recommendation approach. It directly modifies the recommendation score of items by incorporating their timeliness factor. We find that with this modification, the recommendation accuracy is largely improved.

Our work opens up a couple of questions for future research. One straightforward extension would be systematically examine the recommendation performance of all the existing recommendation algorithms with the temporal data division. This will give us a better understanding of the true performance of these methods. Furthermore, some better way to incorporate the timeliness information in the recommendation algorithms can be designed in the future. For instance, a preferential diffusion could be designed in which the diffusion score are biased to the timely items. Last but not least, as the temporal data division include many new items, a more fundamental problem in this situation is the cold-start problem. Even though in the literature many methods have been proposed to solve the cold-start problem. These methods have to be reexamined with the temporal data division for the true effectiveness. Only those with high accuracy in the temporal data division can eventually enjoy a satisfactory performance in real applications.

Acknowledgement

This work is supported by the National Natural Science Foundation of China (Nos. 61603046 and 71361012), the Natural Science Foundation of Beijing (No. 16L00077) and the Jiangxi province Natural Science Foundation (No. 20161BAB201029).

References

Adomavicius, G., Sankaranarayanan, R., Sen, S., & Tuzhilin, A. (2005). Incorporating contextual information in recommender systems using a multidimensional approach. *Acm Transactions on Information Systems, 23*(1), 103–145.

Bobadilla, J., Ortega, F., Hernando, A., & Gutirrez, A. (2013). Recommender systems survey. *Knowledge-Based Systems, 46*(1), 109–132.

Bradley, K., & Smyth, B. (2001). Improving recommendation diversity. *Business*, 75–84.

Campos, P. G., Díez, F., & Cantador, I. (2014). Time-aware recommender systems: a comprehensive survey and analysis of existing evaluation protocols. *User Modeling and User-Adapted Interaction, 24*(1–2), 67–119. doi:10.1007/s11257-012-9136-x.

Deng, S., Huang, L., & Xu, G. (2014). Social network-based service recommendation with trust enhancement. *Expert Systems with Applications, 41*(18), 8075–8084.

Ding, Y., & Li, X. (2005). Time weight collaborative ltering. In *the 14th acm international conference on information and knowledge management* (pp. 485–492).
Feng, W., & Wang, J. (2012). Incorporating heterogeneous information for personalized tag recommendation in social tagging systems. In *Acm sigkdd international conference on knowledge discovery and data mining* (pp. 1276–1284).
Gunawardana, A., & Shani, G. (2008). Evaluating recommender systems. In *International conference on automated solutions for cross media content and multi-channel distribution* (pp. 211–217).
Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS), 22*(1), 5–53. doi:10.1145/963770.963772.
Huang, C. L., Yeh, P. H., Lin, C. W., & Wu, D. C. (2014). Utilizing user tag-based interests in recommender systems for social resource sharing websites. *Knowledge-Based Systems, 56*(C), 86–96.
Jeong, B., Lee, J., & Cho, H. (2010). Improving memory-based collaborative filtering via similarity updating and prediction modulation. *Information Sciences, 180*(5), 602–612.
Konstan, J. A., Miller, B. N., Maltz, D., Herlocker, J. L., Gordon, L. R., & Riedl, J. (1997). Grouplens: applying collaborative filtering to usenet news. *Communications of the Acm, 40*(3), 77–87.
Koren, Y. (2009). Collaborative filtering with temporal dynamics. In *ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 89–97).
Liu, J. G., Zhou, T., & Guo, Q. (2011). Information filtering via biased heat conduction. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics, 84*(3), 1–4. doi:10.1103/PhysRevE.84.037101.
Lü, L., & Liu, W. (2011). Information filtering via preferential diffusion. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics, 83*(6), 1–12. doi:10.1103/PhysRevE.83.066119.
Lü, L., Medo, M., Yeung, C. H., Zhang, Y.-C., Zhang, Z.-K., & Zhou, T. (2012). Recommender systems. *Physics Reports, 519*(1), 1–49.
Martinez-Cruz, C., Porcel, C., J. , B.-M., Moreno, J., & Herrera-Viedma, E. (2015). A model to represent users trust in recommender systems using ontologies and fuzzy linguistic modeling. *Information Sciences, 311*, 102–118.
Massa, P., & Bhattacharjee, B. (2004). Using trust in recommender systems: An experimental analysis. In *the second international conference on trust management, springer berlin* (pp. 221–235).
Qian, X., Feng, H., Zhao, G., & Mei, T. (2014). Personalized recommendation combining user interest and social circle. *IEEE Transactions on Knowledge and Data Engineering, 26*(7), 1763–1777.
Salter, J., & Antonopoulos, N. (2006). Cinemascreen recommander agent: Combining collaborative and content-based filtering. *IEEE Intelligent Systems, 21*(1), 35–41. doi:10.1109/MIS.2006.4.
Salton, G., & Mcgill, M. J. (1983). *Introduction to modern information retrieval*. McGraw-Hill,.
Serrano-Guerrero, J., Herrera-Viedma, E., Olivas, J. A., Cerezo, A., & Romero, F. P. (2011). A google wave-based fuzzy recommender system to disseminate information in university digital libraries 2.0. *Information Sciences, 181*(9), 1503–1516.
Shang, M., Lu, L., Zhang, Y. C., & Zhou, T. (2009). Empirical analysis of web-based user-object bipartite networks. *Epl, 90*(4), 1303–1324.
Song, W., Qiang, G., & Liu, J. (2015). Effect of the time window on the personalized recommendation algorithm. *Complex Systems and Complexity Science, 12*(1), 28–32.
Xiang, L., Yuan, Q., Zhao, S., Chen, L., Zhang, X., Yang, Q., & Sun, J. (2010). Temporal recommendation on graphs via long- and short-term preference fusion. In *ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 723–732).
Xiao, B., & Benbasat, I. (2007). E-commerce product recommendation agents: Use, characteristics, and impact. *Mis Quarterly, 31*(1), 137–209.
Yu, F., Zeng, A., Gillard, S., & Medo, M. (2015). Network-based recommendation algorithms: A review. *Computer Science, 452*, 192208.
Zeng, A., Chi, H. Y., Medo, M., & Zhang, Y. C. (2015). Modeling mutual feedback between users and recommender systems. *Journal of Statistical Mechanics Theory and Experiment, 2015*(7), P07020.
Zeng, A., Vidmer, A., Medo, M., & Zhang, Y. C. (2014). Information filtering by similarity-preferential diffusion processes. *Epl, 105*(5), e58002.
Zhang, F., & Zeng, A. (2012). Improving information filtering via network manipulation. *Epl, 100*(5), e58005.
Zhang, Q. M., Zeng, A., & Shang, M. S. (2013). Extracting the information backbone in online system. *Plos One, 8*(5). e62624–e62624
Zhang, Y. C., Blattner, M., & Yu, Y. K. (2008). Heat conduction process on community networks as a recommendation model. *Physical Review Letters, 99*(15), 12505–12508.
Zhou, T., Kuscsik, Z., Liu, J. G., Medo, M., Wakeling, J. R., & Zhang, Y. C. (2010). Solving the apparent diversity-accuracy dilemma of recommender systems. *Proceedings of the National Academy of Sciences, 107*(10), 4511–4515.
Zhou, T., Ren, J., Medo, M., & Zhang, Y. C. (2007). Bipartite network projection and personal recommendation. *Physical Review E Statistical Nonlinear and Soft Matter Physics, 76*(2), 70–80.
Zimdars, A., Chickering, D. M., & Meek, C. (2001). Using temporal data for making recommendations. In *Seventeenth conference in uncertainty in artificial intelligence* (pp. 580–588).
Ładẏzyński, P., & Grzegorzewski, P. (2015). Vague preferences in recommender systems. *Expert Systems with Applications, 42*(24), 9402–9411.