AMSE JOURNALS-AMSE IIETA publication-2017-Series: Advances B; Vol. 60; N°1; pp 57-73
Submitted Jan. 2017; Revised March 15, 2017, Accepted April 15, 2017
https://doi.org/10.18280/ama_b.600104

An Effective Recommendation Model using User Access Sequence and Context Entropy

*Xiaoyi Deng, **Feifei Huangfu

* College of Business Administration, Huaqiao University
P.R. China, No.269, Chenghua North Rd., Quanzhou 362021, (londonbell@hqu.edu.cn)
** College of Foreign Languages, Huaqiao University
P.R. China, No.269, Chenghua North Rd., Quanzhou 362021, (huangfu@hqu.edu.cn)

Abstract

Collaborative filtering (CF) is one of the most successful recommendation technologies to cope with information overload problem. Conventional neighborhood-based CF models solely use user/item similarities instead of existing user preferences to form neighborhoods, which prediction accuracy excessively relies on. Besides, customers' interests and demands may vary with contexts in different environment. As a result, the recommendations quality of conventional CF models would suffer. To address these issues, this paper developed an effective hybrid CF model by integrating user access sequence and context entropy. The user access sequence is introduced to mitigate the new user cold-start problem, and the context entropy is introduced for measuring uncertainty degree of user rating under different context and calculating user similarity for gathering the most similar users. Experiments on real-world datasets are carried out to compare our method's accuracy with other three algorithms. The results show our method outperforms other methods and improves recommendation quality effectively.

Key words

Collaborative filtering, user access sequence, context entropy, cold-start, nearest neighbor selection, recommender system

1. Introduction

For decades, recommender systems (RSs) have become the most successful application of personalized recommendation for solving the "information overload" problem. RSs receive information from users about items that they are interested in, and then recommend to them items that may fit their needs [1]. Usually, the core of recommender systems relies on well-known algorithms, collaborative filtering (CF) [2], where there are two primary approaches, neighborhood based model (NBM) and latent factor model (LFM). NBM are dependent on the availability of user ratings information, and product recommendations to a target user based on the relationship between their active neighbors, without relying on any information about the items themselves other than their ratings [1]. By contrast, LFM transforms both items and users into the same latent factor space, and then characterizes each entity with a feature vector inferred from the existing ratings [3]. In LFM, the predictions ratings are denoted by the inner product of the corresponding vector pairs. Therefore, NBM based CF has an advantage in situations where it is hard to analyze the different aspects of the data, such as music, videos and other digital products or services. And, NBM based CF has been developed over decades and widely applied in many RSs and Internet-related fields, such as Amazon, Netflix, and Google News.

Despite its advances, CF suffers from several problems, such as data sparsity, cold-start and lack of context [4]. Data sparsity is common for the user-item ratings matrix to be extremely sparse. It makes conventional CF difficult to select nearest neighbors for identifying similar users or items, and hard to produce accurate predictions or recommendations. Specially, the cold-start problem is also known when a new user or item has just entered the system. New items cannot be recommended until someone rates it, and new users are not likely provided proper recommendations because of the lack of their rating or purchase history. To solve these problems, many different dimensionality reduction approaches have been proposed, such as singular value decomposition (SVD) [5], probabilistic matrix factorization (PMF) [6] and collaborative topic regression (CTR) [7]. However, useful information for recommendations related to those approaches may get lost and recommendation quality may be degraded, when certain users or items are discarded [8].

What's more, the customers' interests and demands usually change along with the time in e-commerce, in other words, the interests and demands of customers are context dependent [9]. Therefore, accurate prediction of consumer preference undoubtedly depends upon the degree of the relevant contextual information. Thereby, it is important to incorporate the contextual information of the user's decision scenario into the recommendation process. More specifically, a user might prefer to read stock market report in the evening on weekdays, and on weekends he or she might prefer to do shopping or read comments on movies/TVs. However, the conventional CF methods typically employ constant historical ratings data to determine appropriate recommendations under homogenous context, without taking into consideration of different users' contextual information [8]. These problems severely affected the quality of CF recommendation.

This paper attempts to develop a hybrid CF model, called ASCECF, which takes advantage of user access sequence and context entropy for the sake of improving prediction quality. In our model, the user access sequence is utilized to mitigate the new user cold-start problem, in which new users have browsed some items and presented few vote on items. It is assumed that two items are similar when they share similar user access sequences among multiple users. On the other hand, the entropy context is introduced to measure the uncertainty degree of the user rating behaviors under different contexts, and the uncertainty can be interpreted as how users understand rating domain to distinguish their tastes. Then, two different models is incorporated into the rating prediction process to enhance prediction quality.

The rest of the paper is organized as follows. In section 2, basic CF recommendation approaches and the critical aspects of CF approaches are reviewed. Section 3 introduces and explains the user access sequence based models, context entropy based models and the hybrid model. Then, experimental results are demonstrated and discussed in Section 4. At last, the paper is concluded and future research direction is given in Section 5.

2. Background and Related Work

2.1 Collaborative Filtering

CF generates recommendations based on the data that store how users rated items [2]. To provide recommendations, CF tries first to search for users who have rated the same or similar items. Once the users with common tastes are found, CF will recommend the items highly rated by those users. Generally, the more items that users have rated, the more similar the users are. The procedure of CF can be stated as follows.

It is assumed that $U=\{u_i|i=1,2,\ldots,m\}$ is a set of m users and $I=\{I_j|j=1,2,\ldots,n\}$ is a set of n distinct items. The set of user ratings is denoted by $R=\{(u_i, I_j)|u_i\in U, I_j\in I\}$ which is a $m\times n$ matrix, as shown in equation (1).

$$R=\left(r_{u_i,I_j}\right)_{m\times n},\quad r_{u_i,I_j}=\begin{cases} S \text{ if } u_i \text{ rated } I_j \\ \varnothing \text{ if } u_i \text{ not rated } I_j \end{cases} \tag{1}$$

where $r_{u,I}$ is the rating of the item I by user u, which indicates the user's preference for different items. Usually, $r_{u,I}$ is equal to a real number denoted by S ($S\neq\varnothing$). When $r=\varnothing$, it means that user u_i does not rate a certain item I_j.

After the data preparation, CF needs to select a similarity function to measure how similar two users are. Two of the most well-known similarity measures are Cosine-based similarity and Pearson correlation coefficient [1] defined in equations (2) and (3).

$$Sim_{COS}\left(I_i, I_j\right) = \frac{\sum_{i \in I(u_i,u_j)} r_{u_i,I} \cdot r_{u_j,I}}{\sqrt{\sum_{i \in I(u_i,u_j)} {r_{u_i,I}}^2} \cdot \sqrt{\sum_{i \in I(u_i,u_j)} {r_{u_j,I}}^2}} \tag{2}$$

$$Sim_{Pearson}\left(I_i, I_j\right) = \frac{\sum_{i \in I(u_i,u_j)} (r_{u_i,I} - \overline{r}_{u_i})(r_{u_j,I} - \overline{r}_{u_j})}{\sqrt{\sum_{i \in I(u_i,u_j)} (r_{u_i,I} - \overline{r}_{u_i})^2} \sqrt{\sum_{i \in I(u_i,u_j)} (r_{u_j,I} - \overline{r}_{u_j})^2}} \tag{3}$$

where $r_{u,I}$ is the rating of item I by user u; $\overline{r}_u$ is mean rating of user u, and $I(u_i, u_j)$ represents the items co-rated by users u_i and u_j.

Once similarity calculation has been done, prediction of a rating of an item I_j by user u_i can be obtained for conventional CF methods [5], as shown in the following equation (4).

$$PR\left(u_i, I\right) = \overline{r}_{u_i} + \frac{\sum_{j \in I(u_i,u_j)} Sim\left(u_i, u_j\right) \cdot \left(r_{u_i,I} - \overline{r}_{u_j}\right)}{\sum_{j \in I(u_i,u_j)} \left|Sim\left(u_i, u_j\right)\right|} \tag{4}$$

2.2 Related Work

The conventional CF approaches predict the rating of items for target users only based on the user-item rating matrix. Although CF is a very successful recommending technology, there are still some potential problems: data sparsity, cold-start and lack of context.

To solve the sparsity problem, many different methods have been proposed, such as dimensionality reduction, graph theory and so on. Ben-Shimon et al [5] suggested an ensemble method for the retrieval of the top-N recommendations using the SVD for reducing the recommendation time and improves the predictive performance. Liu et al [6] propose list-wise probabilistic matrix factorization, ListPMF, which maximize the log-posterior over the predicted preference order with the observed preference orders and is able to get recommendation results more consistent with the user preferences. Wu et al [7] supposed the users' ratings on items are affected both by the personal tastes and their trusted friends' favors, and extended CTR with trust ensemble principle for item recommendation in social media. Anand and Bharadwaj [10] proposed various sparsity measure schemes based on local and global similarities for achieving quality predictions.

Due to the extreme situation of data sparsity, i.e. cold-start problem, various works have been proposed, which can be divided into three categories: (1) make use of additional data sources, (2) choose the most prominent groups of analogous users, and (3)enhance the prediction using hybrid methods. For instance, Chen et al [11] employed exiting users' history data and their relationships in the social community to construct an ontology decision model for assisting the recommendation in the cold-start problem. Ahn [12] focused on the existing similarity measures and applied a heuristic similarity measure method that can provide greater value to a similarity for ratings and improve the recommendation performance under the cold-start conditions. Aharon et al [13] introduced a recommendation approach based on latent factor analysis, called OFF-Set, which is able to model non-linear interactions between pairs of features and updates its model per each recommendation-reward observation in a pure online fashion.

These previous researches have made several improvements on conventional CF algorithms, and they partially reduced the effect of data sparsity on the rating prediction. However, there are still some drawbacks, such as additional data are sometimes not available; it is difficult to choose the optimal number of groups and the splitting criteria is worth considering; and irrelevant users are still included in the computation of similarities [14]. Some other researchers have made use of information entropy to improve the recommendation performance under the cold-start condition. For instant, information entropy has been integrated with selective predictability to estimate the relationships between the target users and active users [15]. And, both the entropy of user and item are taken into account for the measurement of the relative difference between user ratings [16]. But, all the rating differences and user differences are usually treated individually without considering the correlation between them when the similarity is computed.

Besides, the accuracy of predicting consumer preference depends on the degree to which the relevant contextual information is integrated into a recommendation model. The context has been recognized as an important factor for personalized recommendation [17]. Mallat et al [18] suggested that the recommendation systems should have the ability to react on demands posed by users' needs under different contexts. Due to the drawback of conventional CF without considering the context, Adomavicius et al [9] proposed a method for prediction in a multidimensional space including contextual information. The dataset is divided to dimensions for different contexts, and the recommendation is defined by selecting a certain kind of context. Shi et al [19] proposed a new context-aware recommendation approach based on tensor factorization for Mean Average Precision maximization (TFMAP) that is designed to work with implicit user feedback and contextual information.

These studies divide users into different groups by using user profiles or items' categories, then use conventional CF to provide recommendation. In despite of the context factor, users' environment that may affect the process of recommending is not considered. These problems severely affected the quality of CF recommendation.

3. Hybrid Model based on User Access Sequence and Context Entropy

In this research, a novel hybrid approach is proposed to provide an accurate recommendation incorporating user access sequence and context entropy with NMB based CF. The proposed recommendation model is composed of two model: user access sequence based prediction and context entropy based prediction.

In the first model, user access sequence is employed for similarity measurement to search target users' nearest neighborhoods and reduce the impact of cold-start problem on prediction quality. The context entropy based model make use of entropy theory to measure the contextual uncertainty of users rating behaviors in RSs, and the context-entropy are utilized for calculating user similarity to form better neighborhoods. At last, the recommendations are made by a hybridization strategy which combines the prediction values of the user access sequence based prediction and the context entropy based prediction.

3.1 User Access Sequence based Model

In daily life, when new users who have registered and presented few votes enter an e-commerce website, they cannot receive any personalized recommendations based on traditional CF technology. Therefore new users may feel that the recommender system does not offer the service they expected, and they may stop using it. Actually, these users may usually browse some product pages which can reflect the interests or preferences of these users. This kind of browsing behaviours will probably play an important role in producing recommendations for new users. Therefore, some researchers made analysis of the user browsing path by using clustering methods [20]. However, if the same user accesses the same product in different way (directly or indirectly), it will lead to a different browsing path which may result in different recommendations generated by these approaches. Meanwhile, operators of e-commerce websites are most concerned about which products had been purchased or browsed by the users, which can be employed to analyze the users' preferences. Apparently, user access sequence is more important than user browsing path in CF-based recommender system.

3.1.1 Definition of User Access Sequence

In this section, user access sequence is introduced to mitigate the new user cold-start problem, in order to enhance prediction quality in CF-based recommendation. The user access sequence is based on the assumption that two products are similar in human mind when they share similar access sequences among multiple users. For example, user access sequences on ten different items by seven users are shown in Table.1.

Table.1. An example of user access sequences

	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8	i_9	i_{10}
u_1	1	1	1	1	0	1	0	0	0	1
u_2	1	0	0	0	1	1	1	0	1	0
u_3	0	1	0	0	1	1	1	0	1	0
u_4	1	1	1	1	1	0	0	0	0	1
u_5	1	1	1	1	0	0	0	0	0	1
u_6	0	0	0	0	1	1	1	0	1	0
u_7	0	0	0	0	1	1	1	0	0	1

In Table.1, the number 1 represents a certain item is accessed by a corresponding user while the number 0 means not. For example, it is clear that items (i_1, i_2, i_3, i_4, i_{10}) look the same from the views of u_1, u_4 and u_5; items (i_5, i_6, i_7 and i_9) are similar because they are accessed by u_2, u_3 and u_6; and the itemset (i_5, i_6, i_7) is a user access sequence from u_2, u_3, u_6 and u_7.

The analysis of user access sequence is stated as follows. It is assumed that m users are denoted by set $U = \{u_i|i = 1, 2,\dots, m\}$, n distinct items are denoted by set $I = \{I_j|j =1, 2,\dots, n\}$, and user access sequence is marked as set $S(u)$, whose lengths are denoted by $|S(u)|$ as shown in equation (5), where I^{u_i} denotes the items accessed by u_i.

$$S(u)=\left\langle u_i,\left\{I^{u_i}\left|I_j^k \in I\right.\right\}\right\rangle,\ \left|S(u)\right|=k \tag{5}$$

User access sequence is a unidirectional growing sequence, and it can be decomposed into a plurality of subsequences with different lengths. The subsequence is denoted by $S^k(u)$, defined in equation (6).

$$S^k(u)=I_j^1 \rightarrow \cdots \rightarrow I_n^k\ ,\ 1 \le k \le n \tag{6}$$

where k indicates the length of subsequence, and j is the ordinal of an item in I. When k = 1, $S^1(u)$ represents a certain item accessed by u; when $1 < k < n$, $S^k(u)$ is called k-length subsequence; when $k = n$, $S^n(u)$ is the user access sequence of u, $S(u)$.

For example, the user access sequence of u_3 in Table 1 is $I_2^1 \rightarrow I_5^2 \rightarrow I_6^3 \rightarrow I_7^4 \rightarrow I_9^5$, and the length of $S(u_3)$ is 5. All subsequences of $S(u_3)$ are shown in Table.2.

Table.2. All subsequences of user access sequence $S(u_3)$

$S^1(u_3)$	$S^2(u_3)$	$S^3(u_3)$	$S^4(u_3)$	$S^5(u_3)$

I_2^1	$I_2^1 \to I_5^2$	$I_2^1 \to I_5^2 \to I_6^3$	$I_2^1 \to I_5^2 \to I_6^3 \to I_7^4$	$I_2^1 \to I_5^2 \to I_6^3 \to I_7^4 \to I_9^5$
I_5^2	$I_5^2 \to I_6^3$	$I_5^2 \to I_6^3 \to I_7^4$	$I_5^2 \to I_6^3 \to I_7^4 \to I_9^5$	
I_6^3	$I_6^3 \to I_7^4$	$I_6^3 \to I_7^4 \to I_9^5$		
I_7^4	$I_7^4 \to I_9^5$			
I_9^5				

3.1.2 User Access Sequence based Nearest Neighbor Selection

When user access sequence and corresponding subsequences are obtained, user access sequence is employed to measure user similarity. For two different users, the measurement of their similarity is based on not only their user access sequence, but also the corresponding subsequences of user access sequence. In other words, both the items accessed by two users and the intersections of their subsequences should be considered when calculating similarity. It is assumed that the user access sequences of users u_i and u_j are respectively denoted by $S(u_i)$ and $S(u_j)$, $|S(u_i)| = m$, $|S(u_j)| = n$; and the subsequences of $S(u_i)$ and $S(u_j)$ are denoted as $\bigcup_{1\le k\le n} S^k(u_i)$ and and $\bigcup_{1\le k\le n} S^k(u_j)$ respectively. Therefore, the user similarity should be measured by k-length subsequences $\bigcup_{1\le k\le n} S^k(u)$ and full-length sequences $S(u)$, respectively.

First of all, the similarity measure of k-length subsequences $\bigcup_{1\le k\le n} S^k(u)$ is introduced. Let $Sim\left(S^k(u_i), S^k(u_j)\right)$ be the similarity of k-length subsequences on $\bigcup_{1\le k\le n} S^k(u_i)$ and $\bigcup_{1\le k\le n} S^k(u_j)$; let the S^k denote the union of $\bigcup_{1\le k\le n} S^k(u_i)$ and $\bigcup_{1\le k\le n} S^k(u_j)$. Assumed that the length of S^k is l, i.e., $|S^k| = l$, the i-th ($1\le i\le l$) subsequences of S^k are denoted by S_i^k, and $S_i^k(u)$ is the ratio of subsequences in $\bigcup_{1\le k\le n} S^k(u)$ containing S_i^k. An $l \times 2$ matrix $M_{u_i,u_j}(k,2)$ is used to store the similarity value between u_i and u_j on a k-length subsequence $\bigcup_{1\le k\le n} S^k(u)$. According to $M_{u_i,u_j}(k,2)$, two vectors $\overrightarrow{S_{u_i,k}}$ and $\overrightarrow{S_{u_j,k}}$ are employed to denote $\bigcup_{1\le k\le n} S^k(u_i)$ and $\bigcup_{1\le k\le n} S^k(u_j)$ respectively, shown in equation (7).

$$\begin{aligned} \overrightarrow{S_{u_i,k}} &= \left\{S_{k,1}(u_i), S_{k,2}(u_i), \cdots, S_{k,l}(u_i)\right\} \\ \overrightarrow{S_{u_j,k}} &= \left\{S_{k,1}(u_j), S_{k,2}(u_j), \cdots, S_{k,l}(u_j)\right\} \end{aligned} \tag{7}$$

Therefore, the method of vector similarity measuring can be utilized for the similarity measurement between $\overrightarrow{S_{u_i,k}}$ and $\overrightarrow{S_{u_j,k}}$, i.e. $Sim\left(\overrightarrow{S_{u_i,k}}, \overrightarrow{S_{u_j,k}}\right)$, as defined in equation (8).

$$Sim\left(\overrightarrow{S_{u_i,k}}, \overrightarrow{S_{u_j,k}}\right) = Sim\left(S^k(u_i), S^k(u_j)\right) = \frac{\overrightarrow{S_{u_i,k}} \cdot \overrightarrow{S_{u_j,k}}}{\left\|\overrightarrow{S_{u_i,k}}\right\| \cdot \left\|\overrightarrow{S_{u_j,k}}\right\|},\ 1 \le k \le \min(m,n) \tag{8}$$

The similarity measurement of full-length sequence $S(u)$ is described as follows. For two different users u_i and u_j, their user access sequences are denoted by $S(u_i)$ and $S(u_j)$, and the length of $S(u_i)$ is usually not equal to that of $S(u_j)$, i.e., $|S(u_i)| \neq |S(u_j)|$. However, traditional similarity measurement method such as Manhattan and Euclidean distance cannot be used to calculate the similarity between $S(u_i)$ and $S(u_j)$. In this paper, the Levenshtein distance widely applied in the field of natural language processing is introduced for the measuring similarity between $S(u_i)$ and $S(u_j)$. The similarity measuring procedure is described as follows.

Let vector $\overrightarrow{S_u}$ denote the user access sequence $S(u)$; let S_u^I denote a certain item in $\overrightarrow{S_u}$; let $Sim(S(u_i),S(u_j))$denote the similarity measurement between $S(u_i)$ and $S(u_j)$. Then, a $(m+1)\times(n+1)$ matrix P is constructed to store the Levenshtein distances, as defined in equation (9).

$$P = \left(P_{ij}\right)_{(m+1)\times(n+1)} = \min\begin{Bmatrix} m_{i-1,j}+1 \\ m_{i,j-1}+1 \\ m_{i-1,j-1}+d \end{Bmatrix} \tag{9}$$

where d is an integer variable; If $S_{u_i}^I = S_{u_j}^I$, $d = 0$; else, $d = 1$.

When the matrix P is established, it can be found that the value of $P_{m+1,n+1}$ is equal to the Levenshtein distance between $\overrightarrow{S_{u_i}}$ and $\overrightarrow{S_{u_j}}$. The similarity measurement of user access sequence can be calculated by equation (10). After both k-length sequence similarity and full-length sequence similarity are obtained, the user access sequence based similarity measurement can be figured out by equation (11).

$$Sim\left(S\left(u_i\right),S\left(u_j\right)\right) = \left|\frac{P_{m,n}}{\max\left(m,n\right)} - 1\right| \tag{10}$$

$$Sim_{AS}\left(u_i,u_j\right) = \sqrt{Sim\left(S\left(u_i^k\right),S\left(u_j^k\right)\right)\cdot Sim\left(S\left(u_i\right),S\left(u_j\right)\right)} = \sqrt{\frac{\overrightarrow{S_{u_i,k}}\cdot\overrightarrow{S_{u_j,k}}}{\left\|\overrightarrow{S_{u_i,k}}\right\|\cdot\left\|\overrightarrow{S_{u_j,k}}\right\|}\cdot\left|\frac{P_{m,n}}{\max\left(m,n\right)} - 1\right|} \tag{11}$$

Then, user access sequence based prediction of item I rated by user u_i can be obtained in the following equation.

$$PR_{AS}\left(u_i,I\right) = \bar{r}_{u_i} + \frac{\sum_{j\in I(u_i,u_j)} Sim_{AS}\left(u_i,u_j\right)\cdot\left(r_{u_i,I} - \bar{r}_{u_j}\right)}{\sum_{j\in I(u_i,u_j)}\left|Sim_{AS}\left(u_i,u_j\right)\right|} \tag{12}$$

3.2 Context Entropy based Model

3.2.1 Description of Context

The context has been identified as an essential factor in affecting user's acceptance of RSs. The previous research [17] on service recommendation has suggested that the user profiles (such as age, occupation and etc.) have a significant impact on service selection, as well as on consumer choice of purchase channel and perception that determine choice. Han et al [21] found that context can be organized as a hierarchical directed acyclic graph with size information, which can be used to compute similarity between content and context for personalization. Mallat [18] concluded that the context can be measured as a construct representing both user profiles and the conditions. The former describes personal features that may affect user's preference, and the latter represents conditions that users meet when they conduct commerce in different places and times. Therefore, context in this study is divided into two categories: user profile and environment information.

User profile is denoted by a triple C_u, which consists of user information including age, gender and occupation. And Environment information is denoted by a triple C_s, which consists of three subsets: weather, time and holidays. C_u and C_s are shown in equation (13).

$$\begin{cases} C_u = \langle Age, Gender, Occupation \rangle \\ Age \in \left(A = \{ A_i \mid i = 1, 2, \ldots, 7 \} \right) \\ Gender \in \left(G = \{0, 1\} \right) \\ Occupation \in \left(O = \{ O_i \mid i = 1, 2, \ldots, 20 \} \right) \end{cases}, \quad \begin{cases} C_s = \langle Weather,\ Time\ ,\ Holiday \rangle \\ Weather \in \left(W = \{ W_i \mid i = 1, 2, \ldots, n \} \right) \\ Time \in \left(T = \{ T_i \mid i = 1, 2, 3 \} \right) \\ Holiday \in \left(H = \{0, 1\} \right) \end{cases} \tag{13}$$

where the set of *Age* is composed of 7 distinct sections: under 18, 18~24, 25~34, 35~44, 45~49, 50~55 and older than 56; the set of *Gender* includes only two elements: male and female that are denoted by 1 and 0, respectively; the set of *Occupation* consists more than 20 different occupation, such as teacher, doctor, engineer, student and so on. The *Weather* s is denoted by set *W* which contains *n* kinds of unique weathers; the *Time* is composed of 3 sections of the daytime: morning, afternoon and evening; the *Holiday* is similar to the *Gender* set which includes 0 and 1. If H=1, it is a holiday; otherwise, it is a working day.

Assumed that $U=\{u_i|i=1,2,\ldots,m\}$ is a set of m users, for any user u_i, context can be denoted by set $C_i=<C_{ui},C_{si}>=(A_i, G_i, O_i, W_i, T_i, H_i)$.

3.2.2 Definition of Context Entropy

In Shannon's information theory, the entropy is defined as the expected value of the information contained in each event in a given message. In general, the more uncertain the event is, the more information it will contain. In other words, the entropy is a measure of unpredictability of information content. The entropy is denoted by H, defined as follows.

$$H(X) = -\sum_{i=1}^{n} P(x_i)\log_2\left(P(x_i)\right),\ x_i \in X,\ \sum_{i=1}^{n} P(x_i) = 1 \tag{14}$$

where $P(x_i)$ is the probability of possible events x_i for message X.

In this paper, the context entropy is defined as the uncertainty degree of user preferences under a certain context. In other words, the context entropy is utilized to measure the uncertainty degree of user rating behaviors under different contexts. The context entropy is denoted by $H_c(I)$ shown as follows:

$$H_C(u,I) = -\sum_{i=1}^{n} P_C(I_i)\log_2(P_C(I_i)), I_i \in I$$
$$P_C(I_i) = \frac{Number\ of\ rating\ on\ I_i}{Total\ number\ of\ rating\ on\ I} \times \frac{Types\ of\ context\ on\ I_i}{Total\ types\ of\ context},\ \sum_{i=1}^{n} P_C(I_i) = 1 \tag{15}$$

where $P_c(I_i)$ represents the occurrence probability of each rating value of a user u on I_i under context C. In general, the smaller the context entropy is, the more certain user preferences are; otherwise, the more uncertain user preferences are.

In order to make the comparison of users having different number of ratings easier, the value of context entropy is normalized into the interval [0,1]. And, the value of context entropy varies inversely as the uncertain degree of user preferences. Therefore, the normalized context entropy can be calculated in equation (16).

$$H_C^n(u,I) = 1 - \frac{H_C(u,I)}{Total\ number\ of\ rating\ on\ I}, I_i \in I$$
$$P_C(I_i) = \frac{Number\ of\ rating\ on\ I_i}{Total\ number\ of\ rating\ on\ I} \times \frac{Types\ of\ contexts\ on\ I_i}{Total\ types\ of\ contexts},\ \sum_{i=1}^{n} P_C(I_i) = 1 \tag{16}$$

3.2.3 Context Entropy based Nearest Neighbor Selection

After the context entropy value is obtained, the process of neighbor selection begins. To find the nearest neighbor of user u, the user similarity values between u and other users are computed by using user rating and context entropy.

The similarity of user rating is measured by Pearson correlation coefficient, as shown in equation (17), where $r_{u,I}$ is the rating of item I by user u; $\overline{r}_u$ is the average rating of user u, and $I(u_i ,u_j)$ represents the items co-rated by users u_i and u_j.

$$Sim_{PCC}(u_i,u_j) = \frac{\sum_{i\in I(u_i,u_j)} (r_{u_i,I} - \overline{r}_{u_i})(r_{u_j,I} - \overline{r}_{u_j})}{\sqrt{\sum_{i\in I(u_i,u_j)} (r_{u_i,I} - \overline{r}_{u_i})^2}\sqrt{\sum_{i\in I(u_i,u_j)} (r_{u_j,I} - \overline{r}_{u_j})^2}} \tag{17}$$

Then, the user similarity is calculated based on $Sim_{UR}(u_i,u_j)$, and the context entropy is regarded as the rating weight of user, as shown in equation (18).

$$Sim_{CE}\left(u_i,u_j\right)=\frac{\sum_{i\in I(u_i,u_j)}\left(H_C^n\left(u_i,I\right)r_{u_i,I}-\overline{r}_{u_i}\right)\left(H_C^n\left(u_j,I\right)r_{u_j,I}-\overline{r}_{u_j}\right)}{\sqrt{\sum_{i\in I(u_i,u_j)}\left(H_C^n\left(u_i,I\right)r_{u_i,I}-\overline{r}_{u_i}\right)^2}\sqrt{\sum_{i\in I(u_i,u_j)}\left(H_C^n\left(u_j,I\right)r_{u_j,I}-\overline{r}_{u_j}\right)^2}} \tag{18}$$

As the calculation of user similarity has been done, the rating prediction of itemset I by user u_i can be obtained by equation (19).

$$PR_{CE}\left(u_i,I\right)=\overline{r}_{u_i}+\frac{\sum_{j\in I(u_i,u_j)}Sim_{CE}\left(u_i,u_j\right)\cdot\left(r_{u_i,I}-\overline{r}_{u_j}\right)}{\sum_{j\in I(u_i,u_j)}\left|Sim_{CE}\left(u_i,u_j\right)\right|} \tag{19}$$

3.3 Model Integration

This module uses the switching hybridization strategy to combine the prediction values of the user access sequence based prediction and the context entropy based prediction, as shown by using equation (20). The hybrid prediction value takes into account all possible ways to obtain a rating prediction value for an active user who has not rated the target item. The parameters α is used to ensure that a high total prediction rating value is obtained only if prediction rating values of both the user access sequence based and the context entropy based approaches are high.

$$PR\left(u_i,I\right)=\begin{cases}0, & \text{if } PR_{AS}\left(u_i,I\right)=0 \text{ and } PR_{CE}\left(u_i,I\right)=0\\ PR_{AS}\left(u_i,I\right), & \text{if } PR_{AS}\left(u_i,I\right)\neq 0 \text{ and } PR_{CE}\left(u_i,I\right)=0\\ PR_{CE}\left(u_i,I\right), & \text{if } PR_{AS}\left(u_i,I\right)=0 \text{ and } PR_{CE}\left(u_i,I\right)\neq 0\\ \alpha\cdot PR_{AS}\left(u_i,I\right)+\left(1-\alpha\right)\cdot PR_{CE}\left(u_i,I\right), & \text{if } PR_{AS}\left(u_i,I\right)\neq 0 \text{ and } PR_{CE}\left(u_i,I\right)\neq 0\end{cases} \tag{20}$$

4. Experiments and Results

In this section, numerical experiments are designed to test and evaluate ASCE-CF. The experiments on three real-world datasets are carried out on a computer with Intel E5-4650 2.7GHz CPU, 32GB RAM and Windows 2008 operation system.

4.1 Experiments Design

All the experiments are carried out on three real world datasets for completeness and generalization of results, as shown in Table 3. These three datasets are publicly open for research purpose and provided by GroupLens Research Group at University of Minnesota and Netflix Company [4]. The sizes of the three datasets are given in Table.3.

Table.3. Characteristics of three datasets

Dataset	*User*	*Movie*	*Rating*	*Sparsity*
MovieLens-100K	943	1682	100K	6.30%
MovieLens-10M	71567	10681	10M	1.31%
Netflix-100M	480,189	17,770	100M	1.17%

The MovieLens and Netflix datasets datasets provide ratings on movies in the scale of 1 to 5, and all timestamp in datasets are converted into standard time format. For all the experiments, all datasets are split into ten disjoint parts for employing the cross validation. Each part is randomly divided into two groups: 80% of data is used as training set and 20% of data is used as test set.

To evaluate the performance of our approach, two different metrics are selected, Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) MAE is the most widely used metric for measuring the deviation of predictions generated by RSs from the user rating. The lower MAE is, the better prediction performance is. RMSE is a statistical accuracy metric representing the accuracy of predicted rating, an important metric for customers (The lower the better). MAE and RMSE are defined in equation (21).

$$MAE = \frac{\sum_{i=1}^{N} |P_i - Q_i|}{N}, \quad RMSE = \sqrt{\frac{\sum_{i=1}^{N} (P_i - Q_i)^2}{N}} \tag{21}$$

To compare the performance of our algorithm, three other CF algorithms are implemented: an item-based CF algorithm (denoted by KNN) [2], an entropy-based CF approach based on item rating prediction (PECF) [15], and a context-aware CF recommendation model (TFMAP) [19]. KNN applies Cosine-based similarity to predict rating; EBCF employs Pearson correlation coefficient and entropy-driven similarity to perform rating prediction; and TFMAP utilizes context-based similarity to predict user rating. Our proposed ASCECF is evaluated by comparing with the three benchmark algorithms, with the parameters α set at 0.5

4.2 Experiments Results

The experimental results from MAE/RMSE comparisons of four algorithms on MovieLens-100K, MovieLens-10M and Netflix-100M are shown in Fig.1 to Fig.3, respectively.

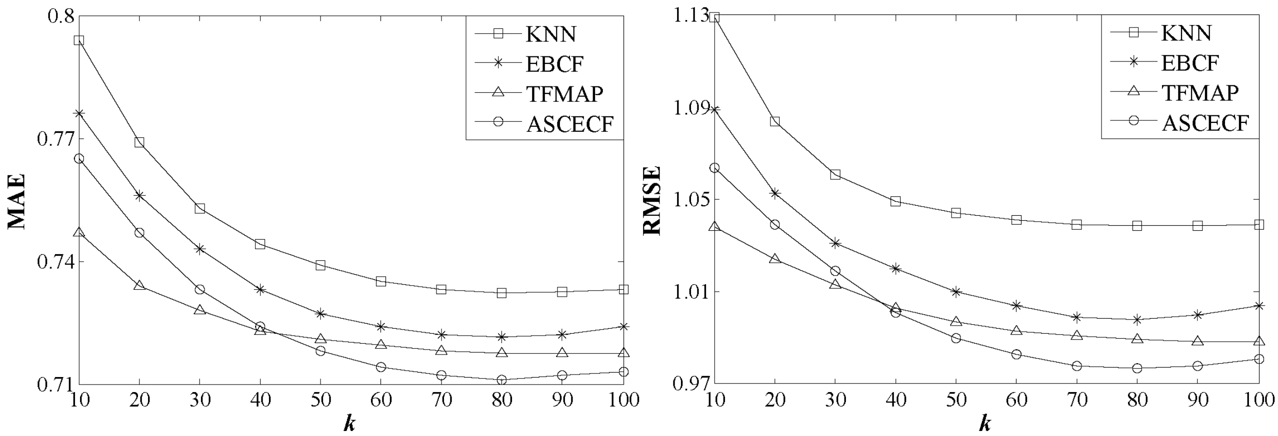

Fig.1. Comparisons of four algorithms results on MovieLens-100K

In Fig.1, the MAE and RMSE values of four algorithms on MovieLens-100K are presented respectively. On one hand, the minimum MAE value of ASCECF is 0.7112 with *k*=80. Both KNN and EBCF obtain their best accuracy values as 0.7323 and 0.7215 with *k*=80, respectively. And TFMAP reach its lowest MAE as *k*=90.On the other hand, the lowest RMSE value of ASCECF is 0.9698 with *k*=80. KNN and EBCF acquire their lowest RMSE values as 1.0386 and 0.998 with *k*=80, respectively, while TFMAP get the most accuracy (0.9883) when *k*=90. The optimal RMSE of ASCECF is 94.05%, 97.88% and 98.72% of that of KNN, EBCF and TFMAP, respectively. The results show that ASCECF has the lowest MAE and RMSE on MovieLens-100K, when $k \in [50,100]$.

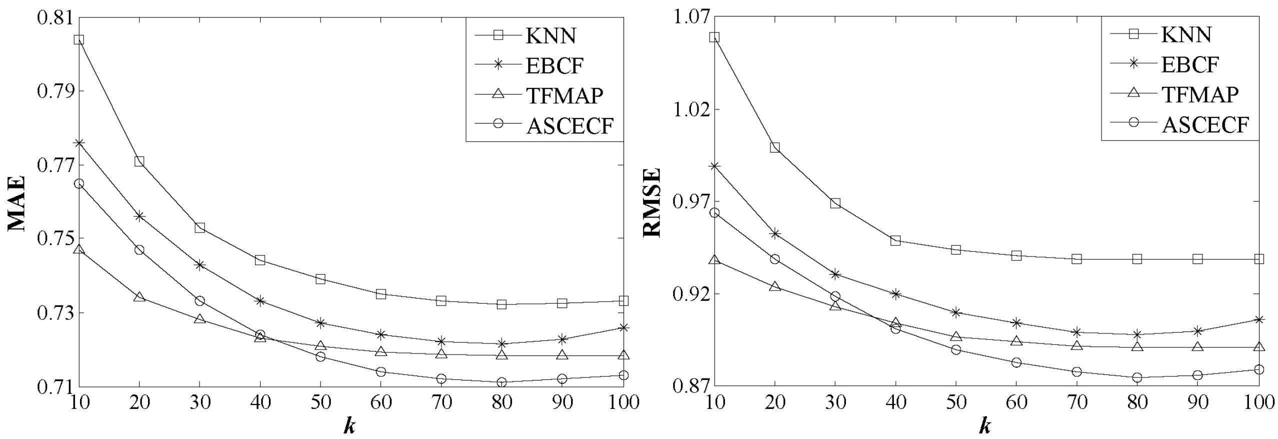

Fig.2. Comparisons of four algorithms results on MovieLens-10M

In Fig.2, the MAE and RMSE values of four algorithms on MovieLens-10M are shown. The minimum MAE values of KNN, EBCF, TFMAP and ASCECF are 0.7398, 0.7219, 0.7184 and 0.7109 as *k*=80, respectively. The MAE value of EBCF and ASCECF is much lower than that of KNN, EBCF and TFMAP. On the other hand, all of four algorithms reach their minimum RMSE values as 0.9386, 0.8983, 0.8914 and 0.875 with k=80 respectively. It is clear that MAE and RMSE of ASCECF are smaller than those of KNN and EBCF with $k\in[50,100]$ on MovieLens-10M.

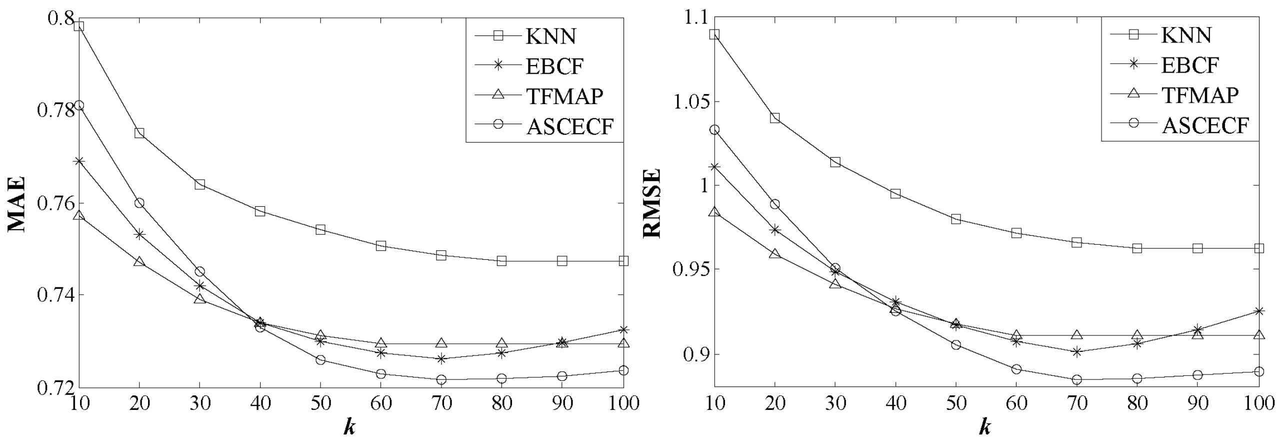

Fig.3. Comparisons of four algorithms results on Netflix-100M

In Fig.3, the MAE and RMSE values of four algorithms on Netflix-100M are shown. The minimum MAE values of EBCF and ASCECF are 0.7261 and 0.7218 with k=70, respectively, KNN gains its minimum MAE as 0.7469 with k=80; and TFMAP reach the MAE trough (0.7296) with k=60. Similar to their MAE results, EBCF and ASCECF obtain their lowest RMSE values as 0.9014 and 0.8845 with k=70 respectively; KNN achieves the highest accuracy with RMSE of 0.9636 as k=80; and TFMAP hit a RMSE trough as 0.9113 with k=60. It is very easy to find that EBCF outperform TFMAP when $k\in[50,80]$, because there is less context information in Netflix-100M than MovieLens datasets. And, ASCECF has better performance than EBCF when integrating user access sequence and context entropy. It is clear that both MAE and RMSE of ASCECF are lower than those of other three methods as $k\geq 50$ on Netflix-100M.

From Fig.1 to Fig.3, it is apparent that our proposed ASCECF not only has the minimum values of RMSE, but also possesses the minimum value of RMSE on three different datasets compared with other three CF methods.

Conclusion

This paper proposed an effective hybrid CF model to improve the recommendation prediction quality. The proposed ASCECF takes advantage of user access sequence for similarity measurement to search target users' nearest neighborhoods and reduce the impact of cold start problem on prediction quality. ASCECF also employs entropy theory to measure the contextual uncertainty of users rating behaviors under different contexts. The context entropy are utilized for calculating user similarity to form better neighborhoods for improving the prediction accuracy. The experimental results have shown that ASCECF succeeds in advancing the prediction quality, which reveals the higher potentials in dealing with the cold-start and context-related issues than conventional CF approaches. In the future, we will integrate uncertainty information of users into trust based CF methods.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (No.71401058), the Program for New Century Excellent Talents in Fujian Province University, NCETFJ (No.Z1625110), and the Project of Science and Technology Plan of Fujian Province of China (No.2017H01010065)

References

1. X. Su, T.M. Khoshgoftaar, A survey of collaborative filtering techniques, 2009, Advances in Artificial Intelligence, vol.12, pp.1-19.
2. B. Sarwar, G. Karypis, J. Konstan, et al, Item-based collaborative filtering recommendation algorithms, 2001, International Conference on World Wide Web, Hong Kong, pp.285-295.
3. Y Koren, R Bell, Advances in collaborative filtering, 2011, Recommender Systems Handbook, Springer, pp.145-186.
4. J. Bobadilla, F. Ortega, A. Hernando, et al, Recommender systems survey, 2013, Knowledge-Based Systems, vol.46, no.1, pp.109-132.
5. D. Ben-Shimon, L. Rokach, B. Shapira, An ensemble method for top-N recommendations from the SVD, 2016, Expert Systems with Applications, vol.64, pp.84-92.
6. J. Liu, C. Wu, Y. Xiong, et al, List-wise probabilistic matrix factorization for recommendation, 2014, Information Sciences, vol.278, pp.434-447.
7. H. Wu, K. Yue, Y. Pei, et al, Collaborative topic regression with social trust ensemble for recommendation in social media systems, 2016, Knowledge-Based Systems, vol.97, pp.111-122..
8. C. He, D. Parra, K. Verbert, Interactive recommender systems: A survey of the state of the art and future research challenges and opportunities, 2016, Expert Systems with Applications, vol.56, pp.9-27,
9. G. Adomavicius, R. Sankaranarayanan, S. Sen, et al, Incorporating contextual information in recommender systems using a multidimensional approach, 2005, ACM Transactions on Information Systems, vol.23, no.1, 103-145.

10. D. Anand, K.K. Bharadwaj, Utilizing various sparsity measures for enhancing accuracy of collaborative recommender systems based on local and global similarities, 2011, Expert Systems with Applications, vol.38, no.5, pp.5101-5109.
11. M. Chen, C.Yang, J.Chen, et al, A method to solve cold-start problem in recommendation system based on social network sub-community and ontology decision model, 2013, The 3rd International Conference on Multimedia Technology, Guangzhou, China, pp.159-166.
12. H.J. Ahn, A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem, 2008, Information Sciences, vol.178, no.1, pp.37-51.
13. M. Aharon, N. Aizenberg, E. Bortnikov, et al, OFF-set: one-pass factorization of feature sets for online recommendation in persistent cold start settings, 2013, The 7th ACM Conference on Recommender systems, Hong Kong, 2013, pp.375-378.
14. H.S. Le, Dealing with the new user cold-start problem in recommender systems: a comparative review, 2016, Information Systems, vol.58, pp.87-104.
15. F. Zhang, H. Liu, Y. Cui, Rating information entropy for cold-start recommendation, 2011, Journal of Information & Computational Science, vol.8, no.1, pp.16-22.
16. W. Wang, G. Zhang, J. Lu, Collaborative Filtering with Entropy - Driven User Similarity in Recommender Systems, 2015, International Journal of Intelligent Systems, vol.30, no.8, pp.854-870.
17. Z.D. Champiri, S.R. Shahamiri, S.S.B. Salim, A systematic review of scholar context-aware recommender systems, 2015, Expert Systems with Applications, vol.42, no.3, pp.1743-1758.
18. N. Mallat, M. Rossi, V.K. Tuunainen, et al, The impact of use context on mobile services acceptance: The case of mobile ticketing, 2009, Information & management, vol.46, no.3, pp.190-195.
19. Y. Shi, A. Karatzoglou, L. Baltrunas, et al, TFMAP: optimizing MAP for top-n context-aware recommendation, 2012, International ACM SIGIR Conference on Research and Development in Information Retrieval, Potland, USA, pp.155-164.
20. A. Said, B. J. Jain, S. Albayrak, Analyzing weighting schemes in collaborative filtering: Cold start, post cold start and power users, 2012, The 27th Annual ACM Symposium on Applied Computing, Trento, Italy, pp.2035-2040.
21. J. Han, H.R. Schmidtke, X. Xie, et al, Adaptive content recommendation for mobile users: Ordering recommendations using a hierarchical context model with granularity, 2014, Pervasive and Mobile Computing, vol.13, pp.85-98.