Information Core Optimization Using Evolutionary Algorithm with Elite Population in Recommender Systems

Caihong Mu[1,2], Huiwen Cheng[1], Wei Feng[1], Yi Liu[1] and Rong Qu[2]
[1]Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education,
International Research Center of Intelligent Perception and Computation, International Collaboration Joint Lab in Intelligent Perception and Computation，Xidian University,
Xi'an, Shaanxi Province 710071, China
[2]ASAP Research Group, School of Computer Science,
University of Nottingham,
Nottingham, UK, NG8 1BB
Emails: mucaihongxd@foxmail.com, Caihong.Mu@nottingham.ac.uk

***Abstract*—Recommender system (RS) plays an important role in helping users find the information they are interested in and providing accurate personality recommendation. It has been found that among all the users, there are some user groups called "core users" or "information core" whose historical behavior data are more reliable, objective and positive for making recommendations. Finding the information core is of great interests to greatly increase the speed of online recommendation. There is no general method to identify core users in the existing literatures. In this paper, a general method of finding information core is proposed by modelling this problem as a combinatorial optimization problem. A novel Evolutionary Algorithm with Elite Population (EA-EP) is presented to search for the information core, where an elite population with a new crossover mechanism named as ordered crossover is used to accelerate the evolution. Experiments are conducted on Movielens (100k) to validate the effectiveness of our proposed algorithm. Results show that EA-EP is able to effectively identify core users, leading to better recommendation accuracy compared to several existing greedy methods and the conventional collaborative filter (CF). In addition, EA-EP is shown to significantly reduce the time of online recommendation.**

Keywords − evolutionary algorithm; elite population; recommender system; core users

I. INTRODUCTION

In an era of big data along with the popularity of the internet, it is becoming more and more difficult and time consuming for people to capture the information and commodities that they are really interested in. In business, recommender system (RS) is utilized to assist users in finding the information they are interested in and provide users with accurate personalized recommendations [1]. By intelligently analyzing historical data of users and their needs, RS provides useful recommendation and helps e-commerce to make profit and gain customers' loyalty.

Collaborative filtering (CF) [2-3], which recommends to the active user (i.e., a user looking for suggestions) the items that other users with similar tastes liked, is considered to be the most popular and widely implemented technique in RS. In general, most existing CF based recommendation algorithms select from all the users L users that are the most similar to the active users, using a similarity measure between the active user and all the other users. This is often too time-consuming, and leads to CF suffer from the scalability limitation. Amatriain et al. [4] proposed a variant of the conventional CF based on expert opinions, where predictions are computed using a set of expert neighbors from an independent dataset rather than applying a nearest neighbor algorithm to the user-rating data. This method is shown to address some of the weaknesses in the conventional CF including scalability and noise in user feedback. They obtained expert ratings by crawling the Rotten Tomatoes2 web site, which aggregated the opinions of movie critics from various media sources. However, such expert ratings are not always available or easy to obtain. Cho et al. [5] made use of experts to generate predictions in a recommender system. In their approach, expert users is identified from a closed community of users according to a derived "domain authority" reputation-like score for each user in the data set. Recently, Zeng et al. [6] presented four greedy methods, namely degree-based, frequency-based, rank-based and random-based method, to find expert users or core users (also named as information core), and found that selecting similar users from a group of core users (which accounts for about 20% of all the users) could obtain the recommendation accuracy up to 91.4% of that from all users in the worst result among different data. A recommender system with these core users sometimes obtains even higher accuracy than that with all users. All above methods are based on a key idea that the historical information from some "expert users" or "core users" is more reliable, objective and positive, thus is key and of higher importance to impact upon the performance of recommender systems. In addition, making recommendations from core users rather than from all the users would reduce the online recommendation time.

Therefore, identifying information core is of great significance for RS to greatly increase the speed of online recommendation, while retain comparable accuracy. However, existing methods only made use of existing expert ratings directly or identified the expert users or core users using some measure defined according to the authors' personal opinion. Different measures may identify different core users. There is no general method to identify core users in the literatures. In this paper, we propose a general method to identify core users by modelling this problem as a combinatorial optimization problem and solve it by a novel evolutionary algorithm (EA).

Assume a recommender system has m users and n items. K users (i.e., core users) are to be selected as information core from the m users ($K<m$) according to some criterion evaluating the performance of recommender systems. Finding information core thus could be modelled as a combinatorial optimization problem.

The combinatorial optimization problem usually could be dealt with by local search algorithm or EAs. Local search algorithms are usually fast and easy to implement [7], thus are suitable for some real world problems including on-line optimization. EAs have the advantage of global search. Considering that the information core usually does not change very often in real world, and off-line optimization can be used, we propose a novel evolutionary algorithm with Elite Population (EA-EP) in this paper to investigate its global optimization in the problem of information core optimization.

Elitism, as a vital important mechanism in evolutionary algorithms, plays an important role in accelerating the speed of convergence of the evolution. Based on the idea of elitism, Mu et al. [8] developed a multiobjective non-dominated neighbor co-evolutionary algorithm (NNCA) with elite population, where the elite individual located in less-crowded region will have more chances to select more team members for its own team and thus this region can be explored more sufficiently. Therefore, the elite population will guide the search to the more promising and less crowded region. NNCA obtained better performance on several benchmark problems about multiobjective function optimization.

In this paper, aiming at solving the single-objective problem of finding information core, we propose a novel mechanism of elite population, i.e., in each generation, all the elite individuals are sorted in a descending order according to their fitness values, and every two neighboring elite individuals in the queue are combined with each other using crossover operator, while the common individuals could not take part in the crossover procedure unless they are improved to become an elite individual by the mutation operator in the following generations. This mechanism ensures that high-quality individuals have more chances to take part in the evolution, to guide the search to the promising regions more quickly. The new mechanism is easier to implement than traditional random crossover, but is more effective, which is shown in the experimental results. In addition, results of experiments show EA-EP can effectively find more relevant core users and obtain much higher accuracy in recommender systems compared to the greedy methods proposed by Zeng et al. [6] and the collaborative filtering method in [3]. Finally, using the core users found by EA-EP for recommendation, the online recommendation time is greatly reduced compared to that of conventional CF.

The contributions of this work are as follows.

(1) The problem of finding core users is modelled for the first time as a combinatorial optimization problem.

(2) A novel evolutionary algorithm with Elite Population is proposed to solve the above combinatorial optimization problem, which is a general method to identify core users, without any extra measurement for identifying the core users.

The remainder of this paper is organized as follows. In section 2, the relevant background is introduced. In section 3, the proposed algorithm is described in detail. In section 4, the experimental results are presented. Conclusion is drawn in section 5 with some future work.

II. BACKGROUND

Existing methods for RS [3] can be divided roughly into three classes: collaborative filtering (CF) based methods, content-based methods and hybrid methods, among which CF methods have showed some great success in industrial applications.

CF methods can be categorized in two categories: model-based and memory-based algorithms. Model-based algorithms generate an offline model and predict ratings online according to the learned model, such as singular value decomposition (SVD) [9], collaborative topic regression (CTR) [10] and so on. Memory-based algorithms make use of users' historical ratings on their purchased items called explicit ratings, or their clicking records called implicit ratings, to calculate the similarity between users according to a similarity measure, and finds from all the users L users that are the most similar to the active user so as to predict possible rating on a specific item or provide a recommendation list according to the predicted ratings. Classic measurements of similarity include Cosine coefficient and Pearson coefficient, etc. Memory-based CF [11-12] algorithms are divided into two classes: user-based and item-based methods. We use the user-based method as the example in this paper. In a CF based RS, assume the rating matrix stores the ratings r_{ui} from the m users for the n items, where r_{ui} indicates the preference by user u of item i. Therefore, to predict the rating r_{ui} that user u rates item i, which is assumed to be unknown, the main procedure of the user-based CF method is as follows.

1) Similarity Computation: Firstly, the similarity S_{uv} between the active user u and any other user v should be calculated using (1) based on the rating matrix in the training data, respectively.

$$S_{uv} = \frac{\sum_{i=1}^{n} r_{ui} \cdot r_{vi}}{\sqrt{\sum_{i=1}^{n} r_{ui}^{\,2}} \sqrt{\sum_{i=1}^{n} r_{vi}^{\,2}}} \tag{1}$$

2) Neighborhood Selection: For each active user u, L users that are the most similar to u according to S_{u*} are found as the neighborhood set N_u.

3) Score Prediction: The rating that the active user u rates item i is predicted by averaging the observed

ratings from the user's neighborhood N_u on item i , and weighted by S_{uv}, as shown in (2). After the score prediction, a recommendation list according to the predicted ratings could be provided to the active user.

$$\hat{r}_{ui} = \frac{\sum_{v \in N_u} S_{uv} \cdot r_{vi}}{\sum_{v \in N_u} S_{uv}} \tag{2}$$

4) Performance Evaluation: To measure the accuracy of the recommendation, the mean average absolute error (*MAE*), which is the most frequently used metric, could be used.

$$MAE = \frac{1}{|I_u|} \sum_{i \in I_u} |\hat{r}_{u,i} - r_{u,i}| \tag{3}$$

Where r_{ui} is the actual rating from user u for item i in the test data, and I_u is the set of total items rated by user u. *MAE* is the average difference between the predicted ratings and actual ratings in the matrix for the test data. The smaller the *MAE* is, the better the algorithm's performance is.

According to the above procedure, it can be seen that most time of CF is spent on the similarity computation, where the similarities between the active user and all the other users are calculated first, and then L neighbors that are the most similar to the active user are selected. If fewer but reliable users can be identified, the similarity computation could be executed on a smaller set of users, and thus reducing the online recommendation time. Our proposed idea is to find this smaller user set, i. e., the set of core users, namely the information core optimization problem.

III. Proposed Algorithm

In this paper, we design an evolutionary with Elite Population (EA-EP) to find the core users, where an elite population is used to accelerate the searching speed.

To find K core users from the m users ($K<m$), one solution or individual of EA-EP could be encoded as $S = (x_1, x_2, ..., x_K)$, where each gene is an integer indicating that the corresponding user is selected as a core user.

In EA-EP, the individual's fitness is defined as the reversed value of *MAE* in (3). Here, the active user's neighbors are obtained based on the core users rather than from all users. For each individual, its core users are decided by its own chromosome, which is in fact one of the candidate solutions for core users. The higher the fitness of the individual, the better the core users indentified by this individual.

In each generation, we select the top 80% individuals in the population as the elite population, sorted in a descending order according to their fitness. An ordered crossover will be applied to the elite population, while the remaining individuals called common population will have no chance to participate in the crossover procedure during the current generation.

Input: The parameters of the algorithm, including the size of the population *PS*, the maximum number of iterations I_{max}, the mutation probability q and the pre-determined number of core users K.

Output: The best solution that has been found and its corresponding fitness in the testing data.

Step 1: Set t = 1 and generate an initial population $P_t = \{S_{t1}, S_{t2}, ..., S_{tPS}\}$ randomly of size *PS*. Each solution $S_{ti} = (x_1, x_2, ..., x_K)$, $i = 1, 2, \cdots, PS$ is generated as a set of positive integers subject to (4).

$$x_j = \lceil l + (h - l) \cdot r_j(0,1) \rceil, \; j = 1, 2, \cdots, K \tag{4}$$

where x_j is the jth variable in solution S_{ti}, $r_j(0,1)$ is a uniformly distributed random value between 0 and 1 generated for x_j, and l and h equal to 0, and the total number of users in the dataset (e.g. 943 in our benchmark), respectively. Set $P_t = P_1$, where P_t is the population of the tth generation.

Step 2: Evaluate the fitness f_{ti} of each solution S_{ti} in population P_t and sort the individuals according to its fitness in a descending order.

Step 3: Select the elite individuals (e.g. top 80% individuals) and carry out the ordered crossover (uniform crossover is used here) on them. To be specific, crossover will be applied to the best individual S_{t1} and the second best individual S_{t2}, and individual S_{ti} will be combined with individual $S_{t(i+1)}$ for crossover, and so on. After crossover, the new generated individuals are added to the common population to form the transition population. Mutation will be applied on the transition population with mutation probability q and *PS* new individuals are generated.

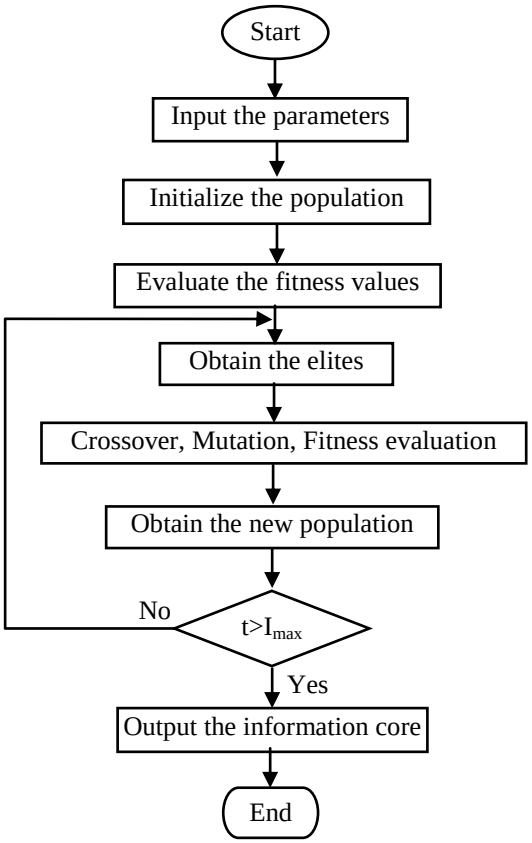

Fig.1 Flowchart of EA-EP

Step 4: The fitness of each new generated individuals will be evaluated, and the best *PS* individuals with the highest fitness will be selected as the next generation P_{t+1} from the combined population including last generation's individuals and the new generated individuals. Set $P_t = P_{t+1}$

Step 5: $t = t + 1$. Repeat step 3 to step 5 till $t > I_{max}$.

Step 6: Output the best solution S_{best} found so far and the corresponding best fitness in the testing data.

In the initial stage and during the evolution, some common users may exist in one set of core user. A simple replacement strategy is used to replace one of the users by another randomly chosen user till there are no common users in one set of core users.

The problem of finding information core is modelled as a combinatorial optimization problem in this paper for the first time, and is solved by the proposed EA-EP, where an elite population with a new crossover mechanism named as ordered crossover is used to accelerate the evolution. The effectiveness of the elite population and the ordered crossover as well as EA-EP based information core optimization will be validated by the following experiments.

IV. EXPERIMENTS

A. Data Set

In order to verify the effectiveness of EA-EP, the Movielens (100k) dataset, a common benchmark data in CF community containing 100,000 explicit ratings on 1682 movies (items) rated by 943 anonymous users, (http://www.grouplens.org/) is used. The ratings range from 1 to 5, and each user rated more than 20 movies. The data, provided by the GroupLens Research Project at the University of Minnesota, are divided into 5 parts for cross validation, each of which include the training data of 80,000 ratings and testing data of 20,000 ratings. To search the group of core users we divide the training data further into two parts. The first part with 60,000 ratings is for selecting core users with EA-EP algorithm and the other part with 20,000 ratings is used to evaluate the performance of core users in terms of *MAE*.

B. Parameter Setting

In the experiments, the population size *PS*, mutation probability *q* and number of core users *K* are set to 100, 0.01, and 159, respectively. The total number of users is 943, and *K* is set as 159, which is about 16.8% of 943 and is a little smaller than 20% which was used in [6]. In addition, the number of neighbors for the active user is set from 10 to 150 with an interval of 10, and every experiment is conducted 10 times, independently. In general, the more the neighbors of the active user are, the better the *MAE* results are. Therefore results with 90, 130, 140 neighbors are shown as representatives in the following experiments, and results with all different number of neighbors are also provided to show the superiority of EA-EP.

C. Experiments Designing and Result.

In order to verify the effectiveness of elite population and ordered crossover, a variant of the EA-EP with a random crossover is tested for comparison, where the crossover probability is set as 0.8, and the value of the population size *PS* and mutation probability *q* are the same as those of EA-EP. Therefore, the numbers of fitness evaluations are the same between two methods with the same maximum generation.

Fig.2 presents the comparative performance of the two variants of EA-EP with 90, 130, 140 neighbors, respectively. As shown in *Fig.2*, EA-EP has a better performance and a faster convergence speed when searching core users, which validates the effectiveness of the strategy of elite population and ordered crossover in EA-EP. Why could EA-EP converge faster? The reason is as follows. In crossover procedure of EA-EP, only elite individuals are allowed to take part in the crossover, and one elite individual only combines with the elite individual whose fitness is the most close to that of the former. Such strategy ensures the excellent genes could be passed or copied to the next generation and thus accelerates the speed of convergence of the algorithm as well as improves the quality of the final solution.

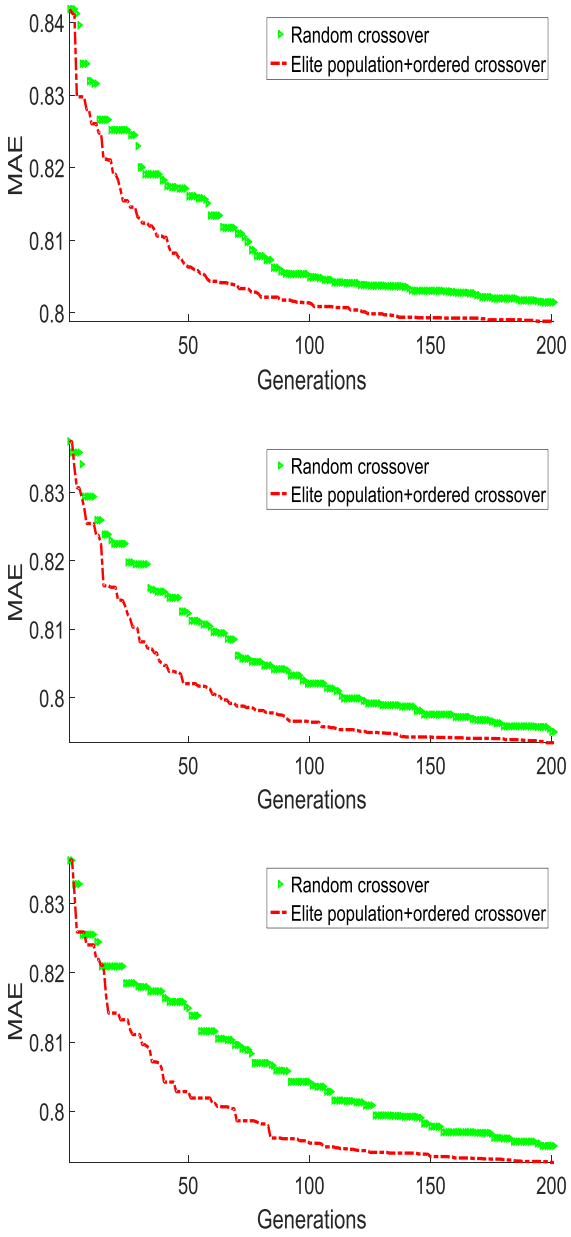

Fig.2 Comparisons between the random crossover and crossover with elitism in EA-EP with 90,130,140 neighborhoods

In order to illustrate the effectiveness of uniform crossover in searching core users we design two other variants of EA-EP, with one point crossover and two point crossover, respectively. All the parameters of two variants are same as those of EA-EP.

Fig.3 shows the results with 90, 130, 140 neighbors, respectively. Results show that the uniform crossover is faster in reaching the promising regions compared to one-point crossover and two-point crossover, demonstrating the superiority of the uniform crossover on the information core optimization problem.

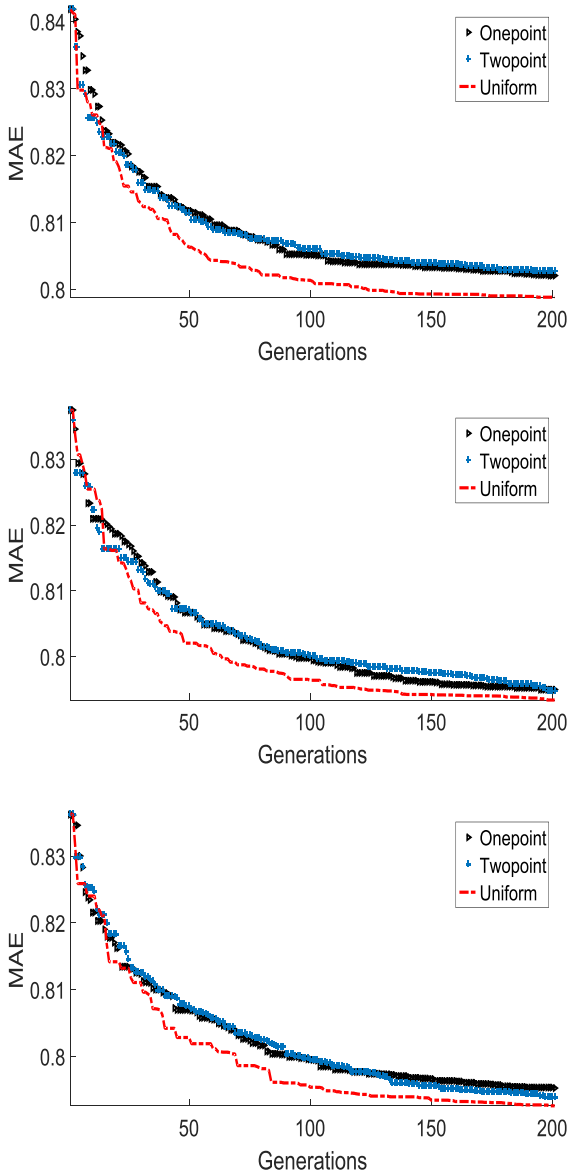

Fig.3 Comparisons between different crossover operators with 90,130,140 neighborhoods in EA-EP

To demonstrate the performance of our proposed EA-EP, we implement the greedy algorithms including the rank-based information core, frequency-based information core, degree-based information core and random-based information core methods proposed in [6] and the conventional collaborative filtering in [3], and compare EA-EP with them. We use the value of *MAE* as the metric to measure the performance of algorithm descripted above.

Fig.4 and Table I present the average value of *MAE* with 90, 130, 140 neighbors, respectively. *Fig.5* shows the average results of *MAE* obtained from random-based, degree-based, frequency-based, rank-based, CF-based method and EA-EP in five cross validations with different number of neighbors. These results show that EA-EP obtained better accuracy compared with both those results from Zeng et al. [6] and the conventional CF method in terms of *MAE*, validating the effectiveness of our proposed EA-EP algorithm.

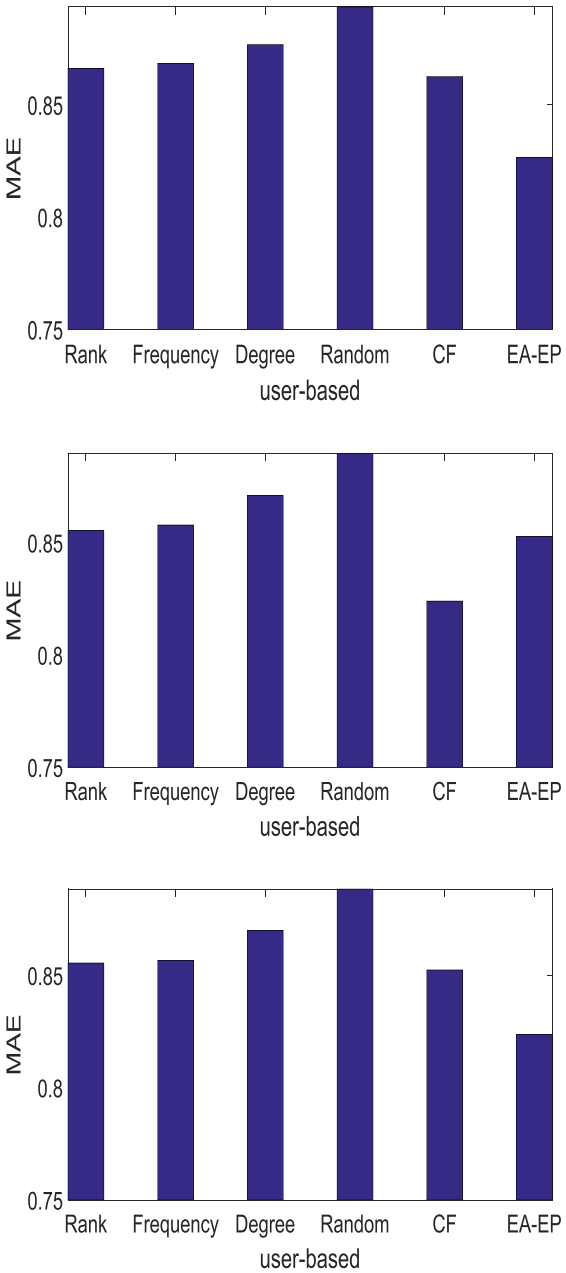

Fig.4 Results of *MAE* with 90,130,140 neighbors from different algorithms

TABLE I Results of *MAE* from different methods with 90,130,140 neighborhoods

	Rank	Frequency	Degree	Random	CF	EA-EP
1	0.883	0.883	0.863	0.884	0.891	0.832
2	0.858	0.860	0.864	0.902	0.900	0.827
3	0.878	0.876	0.877	0.901	0.889	0.822
4	0.854	0.858	0.888	0.886	0.893	0.821
5	0.858	0.864	0.891	0.874	0.881	0.823

	Rank	Frequency	Degree	Random	CF	EA-EP
1	0.868	0.867	0.859	0.884	0.864	0.832
2	0.850	0.854	0.862	0.902	0.848	0.823
3	0.866	0.866	0.874	0.901	0.860	0.824
4	0.841	0.847	0.875	0.886	0.840	0.825
5	0.851	0.856	0.883	0.874	0.850	0.829

	Rank	Frequency	Degree	Random	CF	EA-EP
1	0.866	0.865	0.858	0.884	0.864	0.827
2	0.850	0.853	0.862	0.899	0.847	0.825
3	0.865	0.865	0.873	0.900	0.860	0.820
4	0.841	0.846	0.875	0.885	0.841	0.822
5	0.851	0.855	0.881	0.874	0.850	0.825

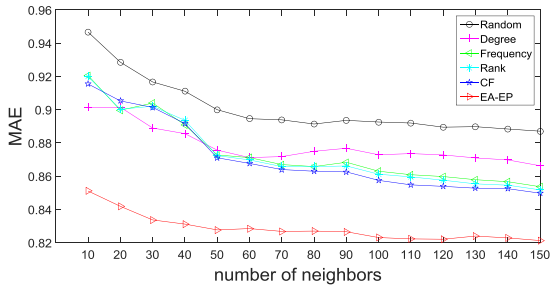

Fig.5 Comparison of average results of MAE in five cross validations with different number of neighbors in different algorithms

To demonstrate the contribution from using the idea of core users in online recommendation compared to the conventional CF, two recommendation algorithms are implemented in machine with Intel Core i5 and 2.50GHz and compared within the CF framework. The difference between them is on obtaining the neighbors of the active user. The former is based on the conventional CF, and obtains neighbors of the active users from all the users, while the latter obtains neighbors of the active users from the core users identified offline using EA-EP.

The average results of online recommendation time cost by two recommendation algorithms with 90, 130 and 140 neighborhoods in 10 independent runs are shown in Table II. It can be concluded that the recommendation algorithms based on the identified core users consume less time compared to the conventional CF in online recommendation. Making use of information from the identified core users greatly reduced the time consumed, contributing to more accurate recommendation to the active users.

TABLE II. Comparison of average online recommendation time between the CF method and the core user method with 90, 130 and 140 neighborhoods in 10 independent runs

neighbors	CF's time(s)	Core users' time(s)
90	160.138	6.092
130	160.642	6.078
140	161.153	6.103

V. Conclusions And Future Work

In this paper, the problem of finding information core is modelled as a combinatorial optimization problem. A novel evolutionary algorithm with Elite Population (EA-EP) is proposed to search the information core. EA-EP makes use of an elite population to accelerate the search speed, and chooses the uniform crossover which is more suitable for the modelled information core optimization problem. Experimental results on the Movielens (100k) dataset show that EA-EP improves the accuracy of recommendation in terms of mean average absolute error (*MAE*), which is the most widely used metric for evaluating the accuracy in recommender systems. In addition, it is shown that core users found offline can also significantly decrease the online recommendation time.

Our future work will introduce advanced machine learning into EA-EP to improve the algorithm's learning ability, which would further accelerate the convergence speed and improve the quality of the final solution when the algorithm is applied to search for the core users.

Acknowledgment

This work was supported by the National Basic Research Program (973 Program) of China (No. 2013CB329402), the National Natural Science Foundation of China (Nos. 61672405, 61373111, 61573015, 61473215 and 61371201), the Fundamental Research Funds for the Central Universities (No. JBG160229), the Fund for Foreign Scholars in University Research and Teaching Programs（the 111 Project）(No. B07048), the Major Research Plan of the National Natural Science Foundation of China (Nos. 91438201 and 91438103), the Program for Cheung Kong Scholars and Innovative Research Team in University (No. IRT_15R53) and China Scholarship Council (CSC).

References

[1] B. Li, C. Qian and J. Li, et al. "Search based recommender system using many-objective evolutionary algorithm," IEEE Congress on Evolutionary Computation. 2016, pp. 120-126.

[2] J. Wei, J. He, and Chen K, et al. "Collaborative filtering and deep learning based recommendation system for cold start items," Expert Systems with Applications, 2016, vol. 69, pp. 29–39.

[3] D. Jannach, M. Zanker and Felfernig A, et al. "Recommender Systems: An Introduction," International Journal Of Human-Computer Interaction, 2010.

[4] X. Amatriain, N. Lathia and J.M. Pujol, et al. "The wisdom of the few: a collaborative filtering approach based on expert opinions from the web," International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2009, pp. 532-539.

[5] J. Cho, K. Kwon and Y. Park. "Collaborative Filtering Using Dual Information Sources," IEEE Intelligent Systems, 2007, vol. 22, no.3, pp. 30-38.

[6] W. Zeng, A. Zeng and H. Liu, et al. "Uncovering the information core in recommender systems," Scientific Reports, 2014, vol. 4, pp. 6140-6140.

[7] B. Chen, R. Qu and R. Bai, et al. "A Variable Neighbourhood Search Algorithm with Compound Neighbourhoods for VRPTW," International Conference on Operations Research and Enterprise Systems. 2016, pp. 25-35.

[8] C. Mu, L. Jiao and Liu Y, et al. "Multiobjective nondominated neighbor coevolutionary algorithm with elite population," Soft Computing, 2015, vol. 19, no. 5, pp. 1329-1349.

[9] H. Polat and W. Du. "SVD-based collaborative filtering with privacy," Proceedings of the 2005 ACM symposium on Applied computing. ACM, 2005. pp. 791-795.

[10] C. Wang, and D.M. Blei. "Collaborative topic modeling for recommending scientific articles," ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, Ca, Usa, August. 2011, pp. 448-456.

[11] N.N. Liu, M. Zhao and Xiang E, et al. "Online evolutionary collaborative filtering," ACM Conference on Recommender

Systems, Recsys 2010, Barcelona, Spain, September. DBLP, 2010, pp. 95-102

[12] K. Yu, A. Schwaighofer and V. Tresp, et al. “Probabilistic Memory-Based Collaborative Filtering,”. IEEE Transactions on Knowledge & Data Engineering, 2004, vol. 16 no.1 pp. 56-69.