An Optimally Weighted User- and Item-based Collaborative Filtering Approach to Predicting Baseline Data for Friedreich's Ataxia Patients

Wenbin Yue, Zidong Wang, *Fellow, IEEE*, Weibo Liu, Bo Tian, Stanislao Lauria and Xiaohui Liu

***Abstract*—In this paper, a modified collaborative filtering (MCF) algorithm with improved performance is developed for recommendation systems with application in predicting baseline data of Friedreich's Ataxia (FRDA) patients. The proposed MCF algorithm combines the individual merits of both the user-based collaborative filtering (UBCF) method and the item-based collaborative filtering (IBCF) method, where both the positively and negatively correlated neighbors are taken into account. The weighting parameters are introduced to quantify the degrees of utilizations of the UBCF and IBCF methods in the rating prediction, and the particle swarm optimization algorithm is applied to optimize the weighting parameters in order to achieve an adequate tradeoff between the positively and negatively correlated neighbors in terms of predicting the rating values. To demonstrate the prediction performance of the proposed MCF algorithm, the developed MCF algorithm is employed to assist with the baseline data collection for the FRDA patients. The effectiveness of the proposed MCF algorithm is confirmed by extensive experiments and, furthermore, it is shown that our algorithm outperforms some conventional approaches.**

***Index Terms*—Friedreich's Ataxia, collaborative filtering, positive correlation, negative correlation, particle swarm optimization.**

I. INTRODUCTION

DURING the past few decades, the recommendation systems (RSs) have received an ever-increasing interest from various communities such as computer science, engineering research and medical applications [1]–[3]. Owing to their outstanding performance in providing users with product or service recommendations, the RSs have found successful applications in a variety of domains including e-commerce, music, movies, news and so on [4]–[6]. In order to recommend goods and services that users are interested in, the RSs mainly employ information filtering technology to analyze users' requirements by mining user behavior data.

Collaborative filtering (CF), as one of the most successful recommendation techniques, has been receiving considerable attention ever since the mid-1990s with fruitful applications in the development of various RSs by Amazon, YouTube, Netflix and so on [7]. Generally speaking, the well-known CF-based recommendation algorithms (RAs) include the user-based CF (UBCF) algorithms and the item-based CF (IBCF) algorithms. The main idea of the UBCF algorithms is to analyze the user behaviors to find similar users (named as neighbors) in the communities. In this case, the items are recommended to a target user based on his/her neighbors' interested items. Similarly, the IBCF algorithms make use of the similarity between the items rather than users. The items that are similar to those in which the target user is interested are recommended to the concerned user.

It should be noticed that the similarity measures play a critical role in the CF-based RAs. Some commonly used similarity measures in the UBCF and IBCF algorithms include the adjusted cosine (AC), cosine, and Pearson correlation coefficient (PCC) measures. Nevertheless, in the case that the user behaviors are complicated, the performance of the CF-based RAs which use the PCC, cosine or AC as the similarity measure cannot be always guaranteed. As such, tremendous efforts have been devoted to the design of more comprehensive similarity measures [8]–[13]. For example, the Shannon entropy has been employed to quantify the users' rating habits [10], [11], where the difference of entropy between users has been utilized as the weight to adjust the result of similarity.

While the state-of-the-art similarity measures have helped improving the prediction accuracy of the RAs, most of the measures take either users or items to predict the missing values. It has been shown in some literature that the combination of the UBCF method and the IBCF method could effectively improve the performance of the RSs [3], [13]–[16]. In [13], the confidence weights, which use the degree of similarity of the neighbors as a reference, have been utilized to balance the predictions obtained by the UBCF method and the IBCF method. In the typical RAs, only positively correlated neighbors are utilized to compute the similarity between the users/items. Nevertheless, the negatively correlated neighbors are also useful in predicting the missing values from another perspective [17]. In this context, a seemingly natural idea is to combine the UBCF and IBCF methods by developing a new prediction model where the positively and negatively correlated neighbors in both methods are taken into account.

To balance the impacts from the UBCF method and the IBCF method, a typical approach is to introduce the weighting parameters to predict the missing values, where the weighting parameters are utilized to make an adequate tradeoff between the positively and negatively correlated neighbors in the UBCF/IBCF methods. It is worth mentioning that, in

This work was supported in part by the Seventh Framework Programme of the European Union under Grant 242193 (EFACTS), the Royal Society of the U.K., and the Alexander von Humboldt Foundation of Germany. (*Corresponding author: Zidong Wang*)

W. Yue, Z. Wang, W. Liu, S. Lauria and X. Liu are with the Department of Computer Science, Brunel University London, Uxbridge UB8 3PH, U.K. (e-mail: Zidong.Wang@brunel.ac.uk).

B. Tian is with the School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China.

the literature, such weighting parameters have been manually selected according to engineering practice by means of certain rules on an ad-hoc basis [3], [13]. Clearly, manual selection of the weighting parameters requires in-depth domain knowledge and specific fine-tuning techniques, which is not always possible in practice. As such, it makes practical sense to automate the parameter selection algorithm with locally optimized performance.

In search of an effective algorithm capable of locating optimally weighted parameters in terms of improving the prediction performance, the Evolutionary computation (EC) algorithms appear to be an ideal candidate. EC algorithms have shown distinguished advantages in solving optimization problems in a diverse range of real-world applications including telecommunication, signal processing, system science and so on [18], [19]. An effective yet popular EC algorithm is the so-called particle swarm optimization (PSO) algorithm that owns the distinctive advantages of easy implementation, quick convergence and great competence in effectively searching the global optimum. So far, the PSO algorithm has gained much attention from both academia and industry with successful applications in solving various multi-objective optimization problems, see e.g. [20], [21]. Owing to its particular suitability, the PSO algorithm is exploited in this paper to optimize the weighting parameters in order to achieve an adequate tradeoff between the positively and negatively correlated neighbors in terms of predicting the rating values.

Motivated by the above discussions, we propose a modified CF (MCF) algorithm in this paper by combining the merits of UBCF and IBCF methods. Through the utilization of the information from both the positively and negatively correlated neighbors, the proposed algorithm is capable of predicting the missing values in multi-aspects with satisfactory accuracy. In particular, the PSO algorithm is dedicatedly exploited to determine (locally) optimized weights of our proposed MCF algorithm so as to further improve the prediction accuracy. To illustrate its application potential, our proposed algorithm is applied to assist with the baseline data collection for Friedreich's ataxia (FRDA) patients. The main contributions are summarized as follows:

1) An MCF algorithm is proposed which not only combines the merits from the UBCF and IBCF methods but also makes full use of the positively and negatively correlated neighbors in predicting the missing values.
2) The PSO algorithm is utilized to optimize the weights in the MCF algorithm so as to achieve a) an adequate tradeoff between the user-based and the item-based similarity measures; and b) a proper balance between the positively and negatively correlated neighbors.
3) The developed algorithm is successfully applied to the FRDA assessment system to assist clinical sample collection for FRDA patients who are unable to attend the tests in the study sites.

The remainder of this paper is structured as follows. The detailed introduction of the proposed MCF approach is presented in Section II. The performance of our proposed MCF approach is evaluated in the case of a real-world neurological disease in Section III. Finally, conclusions are drawn in IV.

II. MAIN RESULTS

Given an RS consisting of m users and n items, the user profiles are denoted by a $m \times n$ matrix called the user-item matrix $R^{m\times n}$. The sets of users and items are defined as $U = \{u_1, u_2, \ldots, u_m\}$ and $I = \{i_1, i_2, \ldots, i_n\}$, respectively. Each element $r_{u,i}$ in R represents that the user u rates the value r on the item i, where $u \in U$, $i \in I$. If the user u has rated the item i, then $r \in 1, 2, \ldots, \tilde{r}$ ($\tilde{r}$ is the upper bound of the ratings). Furthermore, $r_{u,i} = \emptyset$ if the user u does not rate the item i.

A. The computation of similarity

The PCC is one of the most well-known similarity measures in RSs due to its high prediction accuracy and easy implementation [13], [22]. In the UBCF algorithm, the PCC similarity degree between user u and user a is calculated according to the following formula:

$$\mathrm{Sim}_{u,a}^{PCC} = \frac{\sum_{i\in I_{u,a}}(r_{u,i}-\bar{r}_u)(r_{a,i}-\bar{r}_a)}{\sqrt{\sum_{i\in I_{u,a}}(r_{u,i}-\bar{r}_u)^2}\sqrt{\sum_{i\in I_{u,a}}(r_{a,i}-\bar{r}_a)^2}}, \quad (1)$$

where $\mathrm{Sim}_{u,a}^{PCC}$ is the PCC similarity degree between users u and a; $I_{u,a} = I_u \cap I_a$ is the subset of items on which both users u and a have rated, where I_u denotes all the items that have been evaluated by user u and I_a denotes all the items that have been evaluated by user a; $r_{u,i}$ indicates the rating value of item i rated by user u and $r_{a,i}$ indicates the rating value of item i rated by user a; $\bar{r}_u$ is the mean rating value of items that user u has rated; and $\bar{r}_a$ is the mean rating value of items that user a has rated. The values calculated by (1) are in the range of -1 to 1. A larger value of $\mathrm{Sim}_{u,a}^{PCC}$ means that the user u and user a are more similar.

In the IBCF algorithm, the AC method is introduced to evaluate the degree of similarity between the item i and item j by the following formula [23]:

$$\mathrm{Sim}_{i,j}^{AC} = \frac{\sum_{u\in U_{i,j}}(r_{u,i}-\bar{r}_u)(r_{u,j}-\bar{r}_u)}{\sqrt{\sum_{u\in U_{i,j}}(r_{u,i}-\bar{r}_u)^2}\sqrt{\sum_{u\in U_{i,j}}(r_{u,j}-\bar{r}_u)^2}}, \quad (2)$$

where $\mathrm{Sim}_{i,j}^{AC}$ is the AC similarity between items i and j; $U_{i,j} = U_i \cap U_j$ is the subset of users who have rated both item i and item j, where U_i denotes the users who have rated item i and U_j denotes the users who have rated item j; and $r_{u,j}$ denotes the rating value provided by user u on item j. Notice that the values calculated by AC are in the range of -1 to 1.

B. The neighbor selection

Traditionally, the top-k algorithm is used to rank the neighbors based on their similarity degrees in the descending order, and then the top k neighbors are chosen to predict the missing values. As mentioned previously, the values of $\mathrm{Sim}_{u,a}^{PCC}$ and $\mathrm{Sim}_{i,j}^{AC}$ lie in the range of [-1, 1]. The closer that similarity of PCC/AC is to 1, the more similar the users/items are. Users with positive correlations can undoubtedly be used to

make predictions. On the contrary, negative correlation also expresses the relationship between two users from the negative side. The closer that similarity of PCC/AC is to -1, the more dissimilar the users/items are. For example, if users u and a have the similarity of -1, it means when user u rates an item with a high value then user a will definitely give a low value on that item, and vice versa. To sum up, the neighbors with both positive and negative correlations should be utilized to forecast the missing values from different perspectives. The neighbor selection has always been a key yet hot topic in RSs. A large number of neighbor selection strategies have been designed with hope to improve the RSs' performance. Based on the neighbor selection strategy suggested by Breese [22], the neighbors with high correlations are more valuable than those with low correlations. Therefore, the positive and negative neighbor sets of user u and item i are formed by:

$$\text{Pos}_u = \{a^+ | \text{Sim}^{PCC}_{u,a^+} > 0.5, a^+ \neq u\}, \tag{3}$$

$$\text{Neg}_u = \{a^- | \text{Sim}^{PCC}_{u,a^-} < -0.5, a^- \neq u\}, \tag{4}$$

$$\text{Pos}_i = \{j^+ | \text{Sim}^{PCC}_{i,j^+} > 0.5, j^+ \neq i\}, \tag{5}$$

$$\text{Neg}_i = \{j^- | \text{Sim}^{PCC}_{i,j^-} < -0.5, j^- \neq i\}, \tag{6}$$

where Pos_u represents the set of similar users having positive correlation with user u; Neg_u represents the set of similar users having a negative correlation with user u; Pos_i indicates the set of similar items having positive correlation with item i; and Neg_i indicates the set of similar items having negative correlation with item i.

C. The prediction of missing values

In the UBCF methods, the missing values on items are predicted by utilizing positively correlated neighbors of users according to the following formula [22]:

$$\hat{r}_{u,i} = \bar{u} + \frac{\sum_{a^+ \in \text{Pos}_u} \text{Sim}^{PCC}_{u,a^+} (r_{a^+,i} - \bar{a}^+)}{\sum_{a^+ \in \text{Pos}_u} \text{Sim}^{PCC}_{u,a^+}}, \tag{7}$$

where $\hat{r}_{u,i}$ is the predicted value of $r_{u,i}$; $\bar{u}$ is the mean value of different items provided by user u; and $\bar{a}^+$ is the mean value of items provided by the user a^+ who has the positive similarity degree with the target user u. For the UBCF methods that utilize the negative correlation neighbors, the missing values of the test-item are predicted by the following formula:

$$\hat{r}_{u,i} = \bar{u} - \frac{\sum_{a^- \in \text{Neg}_u} \text{Sim}^{PCC}_{u,a^-} (r_{a^-,i} - \bar{a}^-)}{\sum_{a^- \in \text{Neg}_u} \text{Sim}^{PCC}_{u,a^-}}, \tag{8}$$

where $\bar{a}^-$ represents the mean value of items rated by the user a^- who has the negative similarity degree with target user u.

In the IBCF methods employing the positive neighbors, the missing values of the test-items are determined based on

$$\hat{r}_{u,i} = \bar{i} + \frac{\sum_{j^+ \in \text{Pos}_i} \text{Sim}^{AC}_{i,j^+} (r_{u,j^+} - \bar{j}^+)}{\sum_{j^+ \in \text{Pos}_i} \text{Sim}^{AC}_{i,j^+}}, \tag{9}$$

where $\bar{i}$ represents the average values of item i rated by users, and $\bar{j}^+$ is the average value of item j^+ which has the positive similarity degree with the target item i. To be specific, the missing values on items are predicted by utilizing the negatively correlated neighbors according to the following formula:

$$\hat{r}_{u,i} = \bar{i} - \frac{\sum_{j^- \in \text{Neg}_i} \text{Sim}^{AC}_{i,j^-} (r_{u,j^-} - \bar{j}^-)}{\sum_{j^- \in \text{Neg}_i} \text{Sim}^{AC}_{i,j^-}}, \tag{10}$$

where $\bar{j}^-$ is the average value of the item j^- which has the negative similarity degree with the target item i.

In our work, the UBCF method and the IBCF method are combined where both the positively and the negatively correlated neighbors are taken into account to predict the missing values. Three weighting parameters are employed in the developed MCF algorithm in order to achieve 1) a proper balance between the UBCF method and the IBCF method, 2) an adequate tradeoff between the positively and negatively correlated neighbors in UBCF method, and 3) an adequate tradeoff between the positively and negatively correlated neighbors in IBCF method. The formula for prediction is shown as follows:

$$\begin{aligned}
&\hat{r}_{u,i} \\
=&\alpha \times \left(\bar{u} + \lambda \times \frac{\sum_{a^+ \in \text{Pos}_u} \text{Sim}^{PCC}_{u,a^+} (r_{a^+,i} - \bar{a}^+)}{\sum_{a^+ \in \text{Pos}_u} \text{Sim}^{PCC}_{u,a^+}} \right. \\
&\left. -(1-\lambda) \times \frac{\sum_{a^- \in \text{Neg}_u} \text{Sim}^{PCC}_{u,a^-} (r_{a^-,i} - \bar{a}^-)}{\sum_{a^- \in \text{Neg}_u} \text{Sim}^{PCC}_{u,a^-}} \right) \\
&+ (1-\alpha) \times \left(\bar{i} + \beta \times \frac{\sum_{j^+ \in \text{Pos}_i} \text{Sim}^{AC}_{i,j^+} (r_{u,j^+} - \bar{j}^+)}{\sum_{j^+ \in \text{Pos}_i} \text{Sim}^{AC}_{i,j^+}} \right. \\
&\left. - (1-\beta) \times \frac{\sum_{j^- \in \text{Neg}_i} \text{Sim}^{AC}_{i,j^-} (r_{u,j^-} - \bar{j}^-)}{\sum_{j^- \in \text{Neg}_i} \text{Sim}^{AC}_{i,j^-}} \right),
\end{aligned} \tag{11}$$

where α denotes the weight for the UBCF method; $(1-\alpha)$ denotes the weight for the IBCF method; λ and $(1-\lambda)$ represent the weights of the positively correlated neighbors and negatively correlated neighbors in the UBCF method, respectively; β and $(1-\beta)$ denote the weights of the positively correlated neighbors and negatively correlated neighbors in the IBCF method, respectively.

It is worth mentioning that the formula (11) would be degenerated into that for the traditional UBCF algorithm when α and λ are equal to 1, and into that of the traditional IBCF algorithm when $\alpha = 0$ and $\beta = 1$.

D. The PSO-based parameter selection strategy

The PSO algorithm, which is a popular evolutionary computation algorithm inspired by the simulation of the social behavior of fish-schooling/birds-flocking, is applied in this paper to dispose of the parameter optimization problem because of its competitive strength in seeking a relatively satisfactory solution as well as its easy-to-implement feature [20]. Here, each particle in the swarm indicates a candidate solution to the research problem.

In the proposed MCF algorithm, we select three appropriate weighting parameters to guarantee the prediction performance. The weights are expressed by a 3-dimensional vector as follows:

$$\omega \triangleq \begin{bmatrix} \alpha & \beta & \lambda \end{bmatrix}^{\mathrm{T}}.$$

Without loss of generality, we divide the user-item matrix R into the training set (with 60 percent of the data), the validation set (with 20 percent of the data) and the testing set (with 20 percent of the data). The training set is applied to train the weighting parameters, and the validation set is utilized to validate the predicted results by using the trained weighting parameters. As the prediction accuracy reaches the desired threshold, the trained weighting parameters are applied to predict the results in the testing set.

The fitness function of the PSO algorithm is shown as follows:

$$\textit{fitness} = \frac{1}{|V|} \sum_{r_{u,i} \in V} \left| r_{u,i} - \hat{r}_{u,i} \right|, \tag{12}$$

where V represents the validation set, $|V|$ denotes the number of ratings in the validation set and $\hat{r}_{u,i}$ is calculated by formula (11).

Our attention is focused on choosing suitable ω so as to minimize the fitness function of the PSO algorithm. The optimization problem in our work is defined by:

$$\omega^* = \arg\min fitness. \tag{13}$$

In this paper, the particles move at a certain speed in a 3-dimensional search space. Denote

$$v_m(k) = \begin{bmatrix} v_{m1}(k) & v_{m2}(k) & v_{m3}(k) \end{bmatrix}^{\mathrm{T}},$$
$$\omega_m(k) = \begin{bmatrix} \omega_{m1}(k) & \omega_{m2}(k) & \omega_{m3}(k) \end{bmatrix}^{\mathrm{T}},$$

as the velocity and position of the m-th particle at the k-th iteration, respectively. The historical best position of the m-th particle ($m = 1, 2, \ldots, N$) at the k-th iteration and the global best position detected by the entire swarm are, respectively, denoted by

$$p_m(k) = \begin{bmatrix} p_{m1}(k) & p_{m2}(k) & p_{m3}(k) \end{bmatrix}^{\mathrm{T}},$$
$$g(k) = \begin{bmatrix} g_1(k) & g_2(k) & g_3(k) \end{bmatrix}^{\mathrm{T}}.$$

The velocity and the position of the m-th particle are updated by the following equation:

$$\begin{aligned} v_m(k+1) &= w v_m(k) + c_1 r_1 (p_m(k) - \omega_m(k)) \\ &\quad + c_2 r_2 (g(k) - \omega_m(k)), \\ \omega_m(k+1) &= \omega_m(k) + v_m(k+1), \end{aligned} \tag{14}$$

where w is the inertia weight factor; c_1 is the acceleration coefficient called the cognitive parameter, and c_2 is another acceleration coefficient called the social parameter; r_1 and r_2 are two random numbers that satisfy the uniform distribution in the range of 0 to 1; k is the number of current iteration.

In order to enhance the search ability and reduce the possibility of getting trapped into local optima, lots of improved algorithms have been proposed to adjust the parameters in PSO algorithm. In this paper, w is formulated according to the relationship between current iteration and maximum iteration number as mentioned in [24], [25], which is given as follows:

$$w(k) = w_f + (w_i - w_f) \times \frac{k_{\max} - k}{k_{\max}}, \tag{15}$$

where k and $k_{\max}$ are the number of current iteration and maximum iteration, respectively; w_i is the initial inertia weight value when $k = 0$, and w_f indicates the final value of the inertia weight when $k = k_{\max}$.

In this paper, the initial and final inertia weights values are set as $w_i = 0.9$ and $w_f = 0.4$, respectively. In general, a large inertia weight will benefit the global exploration at the early stage and a small inertia weight will help the local exploitation at the later stage. In addition, the acceleration coefficients c_1 and c_2 are calculated by the following equations [26]:

$$c_1 = c_{1f} + (c_{1i} - c_{1f}) \times \frac{k_{\max} - k}{k_{\max}}, \tag{16}$$

$$c_2 = c_{2f} + (c_{2i} - c_{2f}) \times \frac{k_{\max} - k}{k_{\max}}, \tag{17}$$

where c_{1i} denotes the initial value of cognitive acceleration coefficient c_1 and c_{1f} denotes the final value of cognitive acceleration coefficient c_1, c_{2i} denotes the initial value of cognitive acceleration coefficient c_2 and c_{2f} denotes the final value of cognitive acceleration coefficient c_2. According to experiment experience, the values of c_{1i}, c_{1f}, c_{2i} and c_{2f} are set to be 2.5, 0.5, 0.5, and 2.5, respectively. Finally, when the PSO algorithm terminates, we can obtain the optimal parameter vector as $\omega^* = g(k_{\max})$, where $k_{\max}$ represents the number of maximum iteration.

The pseudocode of the MCF algorithm is shown in Algorithm 1 on next page.

Algorithm 1 The MCF Algorithm

Input: User-item rating matrix R, k in top-k method, parameters in the PSO algorithm

1: Divide all the known data in R into the training set and the validation set with a certain proportion;
2: Calculate the PCC similarity between users and the AC similarity between items by utilizing the data in the training set;
3: Employ the PSO technique to select the optimal parameter vector ω^* (presented in Steps 4-14) on the validation set:
4: Initialize velocity and position for each particle;
5: **for** $k = 0$ to $k_{\max}$ **do**
6: **for** $p = 1$ to N **do**
7: Predict the rating values on the validation set based on equation (11);
8: Calculate the fitness value based on equation (12);
9: Obtain $p_m(k)$;
10: **end for**
11: Obtain $g(k)$;
12: Update velocity and position for each particle based on equation (14);
13: **end for**
14: Set $\omega^* = g(k_{\max})$;
15: Calculate the PCC similarity between users and the AC similarity between items by utilizing all the known data;
16: Predict the missing values in R by equation (11) according to the values of α, β, λ in ω^*.

Output: The predictions of missing values in rating matrix R

TABLE I
RATING DATA FORMULATION IN SARA

Gait	Stance	Sitting	Speech disturbance	Finger chase			Nose-finger test		
				right	left	mean [a]	right	left	mean [a]
0 to 8	0 to 6	0 to 4	0 to 6	0 to 4	0 to 4	0 to 4	0 to 4	0 to 4	0 to 4

Fast alternating hand movements			Heel-shin slide			SARA Total [b]
right	left	mean [a]	right	left	mean [a]	
0 to 4	0 to 4	0 to 4	0 to 4	0 to 4	0 to 4	0 to 40

[a] The mean indicates the average value of right and left sides.
[b] The SARA Total indicates the sum of the values on first 4 test-items and the mean values on last 4 test-items.

III. APPLICATION IN FRIEDREICH'S ATAXIA ASSESSMENT SYSTEM

A. FRDA assessment with the help of CF method

Friedreich's ataxia (FRDA), which is defined by a German neurologist in 1863, is an inherited neurodegenerative disorder that affects the nervous system and the heart with symptoms of deep sensory loss, muscle weakness, kyphoscoliosis, dysarthria, heart disease and difficulty in speech [27]. FRDA is the most common hereditary ataxia with 1-2 cases in every 50,000 white people. To comprehensively study FRDA, the European Friedreich's Ataxia Consortium for Translational Studies (EFACTS) has assembled a body of expertise to adopt a translational research strategy for FRDA [28], [29].

EFACTS has been devoted to collecting and analyzing FRDA patient baseline data since 2010. Up to now, EFACTS has collected more than one thousand patients' baseline data from nearly twenty study sites in nine European counties, but the coverage is still far from enough. According to the morbidity rate, the potential FRDA patients are huge. Due to the limitations of physical, psychological or economic reasons, many patients may not be able to go to the study sites for the FRDA medical assessment.

Note that most baseline data are collected through interviews, questionnaires, observations and coordinated tests at the study sites without using any medical instruments. Here, the detailed test methods and rating rules have been provided by EFACTS. Therefore, we make a reasonable assumption that patients who are not able to go to the study sites can be assessed at home and let their families (or themselves) act as examiners. The examiners can be relied upon in providing certain reliable ratings in the portion of test-items during long-term observation and care.

Intuitively, similar FRDA patients exhibit similar symptoms. The unfilled parts in test-items are regarded as missing values. The prediction of missing values can be considered as a typical RS problem, where the patients correspond to the users, and FRDA test-items correspond to the items. Inspired by the idea of CF, the missing values can be predicted by utilizing the certain values provided by the examiners and the data collected by EFACTS. Therefore, the application of our proposed MCF algorithm in FRDA provides an alternative way to assist patient baseline data collection. In this way, many more patient samples can be exploited in clinical trials, which will provide better bases for FRDA research [30], [31].

B. Data pre-processing

In this paper, the scale for the assessment and rating of ataxia (SARA) dataset has been selected from the database provided by the EFACTS. SARA is a new clinical scale that is utilized to evaluate the treatment effectiveness and severity of different types of cerebellar ataxia such as Friedreich's, spinocerebellar and sporadic ataxia [32]. As shown in Tab. I, there are 12 test-items in 8 categories to assess a range of different impairments. The categories are gait, stance, sitting, speech disturbance, finger chase, nose-finger test, fast alternating hand movements and heel-shin slide. SARA has an accumulative score ranging from 0 to 40 where 0 means no ataxia and 40 means most severe ataxia.

The number of patients in the SARA dataset is continuously updated. Up to now, the SARA dataset has included the baseline data of 1029 patients. The user-item matrix R is a 1029×12 matrix, where each row denotes an FRDA patient, and each column denotes a test-item. As shown in Tab. I, the rating intervals are different. Therefore, we normalize the rating values into the 0-1 range based on

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}}, \tag{18}$$

where x' is the normalized value, $x_{\min}$ and $x_{\max}$ are, respectively, the minimum and maximum values of x which give the range of x.

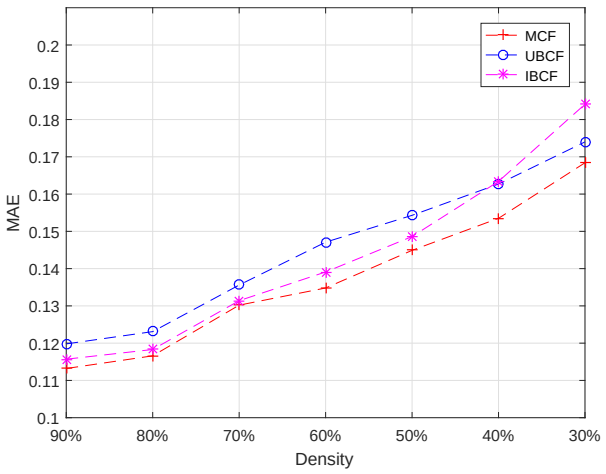

Fig. 1. MAE metric under different densities.

TABLE II
EXPERIMENTAL RESULTS UNDER DIFFERENT DENSITIES

Metrics	Methods	density of matrix						
		90%	80%	70%	60%	50%	40%	30%
MAE	MCF	0.1132	0.1166	0.1302	0.1348	0.1449	0.1535	0.1684
	UBCF	0.1198	0.1231	0.1356	0.1471	0.1543	0.1627	0.1740
	(Improve)	(5.51%)	(5.28%)	(3.98%)	(8.36%)	(6.09%)	(5.65%)	(3.22%)
	IBCF	0.1157	0.1183	0.1314	0.1391	0.1486	0.1634	0.1841
	(Improve)	(2.16%)	(1.44%)	(0.91%)	(3.09%)	(2.49%)	(6.06%)	(8.53%)
RMSE	MCF	0.1583	0.1592	0.1722	0.1769	0.1907	0.1977	0.2265
	UBCF	0.1634	0.1643	0.1811	0.1945	0.2065	0.2209	0.2389
	(Improve)	(3.12%)	(3.10%)	(4.91%)	(9.05%)	(7.65%)	(10.50%)	(5.19%)
	IBCF	0.1601	0.1627	0.1802	0.1845	0.2012	0.2156	0.2431
	(Improve)	(1.12%)	(2.15%)	(4.44%)	(4.12%)	(5.22%)	(8.30%)	(6.83%)

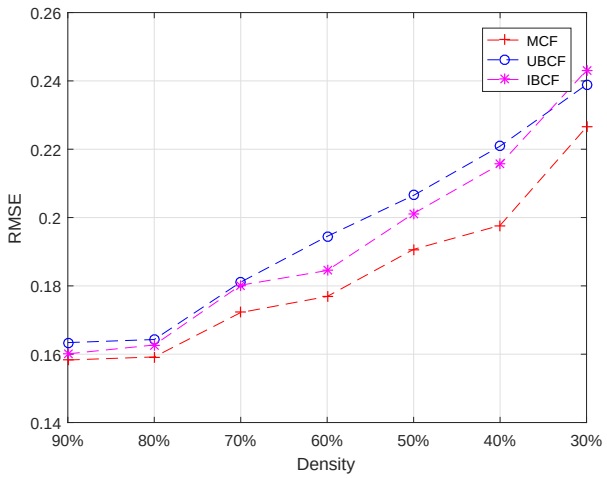

Fig. 2. RMSE metric under different densities.

C. Experiment setting

In our simulation, 1029 patients have been divided into the training set (70%), validation set (15%) and testing set (15%). The training set and validation set are used for selecting the parameter vector ω to minimize the error. The data in the testing set is regarded as patients who cannot take the tests in any study site. In this case, the patients in the testing set only provide ratings on the portion of test-items. The proposed MCF method is utilized to predict the rating values on patients' unfilled parts.

To evaluate the prediction quality of the algorithm, the mean absolute error (MAE) and the root mean square error (RMSE) used in our experiments are given as follows:

$$\text{MAE} = \frac{1}{N}\sum_{u\in U_d}\sum_{i\in I_d}|r_{u,i}-\hat{r}_{u,i}|, \tag{19}$$

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{u\in U_d}\sum_{i\in I_d}(r_{u,i}-\hat{r}_{u,i})^2}, \tag{20}$$

where N represents the total number of predicted values in the testing set; U_d and I_d represent the user set and test-item set in the testing set, respectively; $r_{u,i}$ is the true rating value in the testing set; and $\hat{r}_{u,i}$ is the predicted value provided by our proposed CF algorithm.

The parameters of the PSO algorithm in the simulation are given as follows. The dimension of each particle is 3; the population of the swarm is 20; the maximum iteration number is set to be 1000; and the search space of α, β, λ is in the interval of [0, 1].

D. Results and Discussion

In this paper, we implement our approach on the SARA dataset provided by EFACTS to evaluate the effectiveness of our algorithm by employing the density of the testing set from 90% to 30% with a step size of 10%. We repeat each experiment 100 times to avoid random influence, and the average values of MAE and RMSE have been recorded. To demonstrate the superiority of our proposed MCF algorithm, we make a comparison of the UBCF and IBCF methods with our proposed MCF method on the MAE and RMSE metrics.

Experiment results of the UBCF, IBCF and MCF methods are shown in Figs. 1 and 2. The vertical coordinate denotes the values of MAE or RMSE, and the horizontal coordinate represents the different densities of the user-item matrix. The MAE and RMSE of different CF-based algorithms are displayed in Tab. II. The results indicate that our MCF algorithm has better MAE and RMSE values than the UBCF and the IBCF algorithms under different densities. To sum up, the proposed MCF algorithm has shown satisfactory prediction accuracy in the FRDA baseline data.

E. Complexity Analysis

Classic UBCF (IBCF) algorithm involves the calculation of user-user (item-item) similarity matrix in an offline way, which is computationally expensive. For both UBCF and IBCF, the offline computation of similarity matrices is very time-consuming. The offline time complexity of UBCF and IBCF is $O(m^2 \cdot n)$ and $O(m \cdot n^2)$, respectively, where m denotes the number of users and n denotes the number of items. In MCF, the offline computation is even more expensive because our proposed algorithm needs to compute both user-user and item-item similarity matrices. The offline time complexity of the MCF is $O(m^2 \cdot n + m \cdot n^2)$.

In the online phase, the time complexity of MCF method in the prediction part is the same as that of UBCF/IBCF method, which is $O(k)$ where k is the size of the neighbors of the target user and item. To sum up, our proposed method improves the prediction accuracy at the expense of extra offline computation.

IV. Conclusion

In this paper, an MCF algorithm has been presented and successfully employed to deal with the data prediction problem of FRDA patient baseline data. The proposed MCF algorithm has combined the merits of both the UNCF method and the IBCF method, and has been shown to outperform the UNCF method alone or the IBCF method alone. It should be pointed out that the positively and the negatively correlated neighbors have also been taken into account in the MCF algorithm with hope to improve prediction accuracy. In the developed MCF algorithm, the weighting parameters have been employed to balance the usage of 1) the UBCF method and the IBCF method; and 2) the positively and the negatively correlated neighbors. The PSO algorithm has been applied to automate the selection of locally optimized weights so as to guarantee the prediction accuracy. The MCF algorithm has been applied to deal with a real-world disease, the FRDA, to justify its application potential. Experiment results have shown that our proposed approach greatly improves the prediction accuracy with better performance than either the UBCF algorithm or the IBCF algorithm.

In the future, we aim to explore the possibility of using dynamical systems [33]–[40], deep learning techniques [41]–[46] and up-to-date optimization approaches to improve the developed recommendation systems [47], [48].

References

[1] F. Luo, G. Ranzi, X. Wang, and Z. Y. Dong, Social information filtering-based electricity retail plan recommender system for smart grid end users, *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 95–104, 2017.

[2] Y. Zhang, M. Chen, D. Huang, D. Wu, and Y. Li, iDoctor: Personalized and professionalized medical recommendations based on hybrid matrix factorization, *Future Generation Computer Systems*, vol. 66, pp. 30–35, 2017.

[3] Z. Zheng, H. Ma, M. R. Lyu, and I. King, WSRec: A collaborative filtering based web service recommender system, In: *Proceedings of the 2009 IEEE International Conference on Web Services*, Los Angeles, USA, pp. 437–444, July 6–10, 2009.

[4] X. Luo, M. Zhou, S. Li, D. Wu, Z. Liu, and M.-S. Shang, Algorithms of unconstrained non-negative latent factor analysis for recommender systems, *IEEE Transactions on Big Data*, 2019, DOI: 10.1109/TBDA-TA.2019.2916868.

[5] X. Luo, M. Zhou, S. Li, and M.-S. Shang, An inherently nonnegative latent factor model for high-dimensional and sparse matrices from industrial applications, *IEEE Transactions on Industrial Informatics*, vol. 14, no. 5, pp. 2011–2022, 2017.

[6] Y. Yang, Z. Zheng, X. Niu, M. Tang, Y. Lu, and X. Liao, A location-based factorization machine model for web service QoS prediction, *IEEE Transactions on Services Computing*, 2018, DOI:10.1109/TSC.2018.2876532.

[7] J. Bobadilla, F. Ortega, A. Hernondo, and A. Gutiérrez, Recommender systems survey, *Knowledge-Based Systems*, vol. 46, pp. 109–132, 2013.

[8] F. S. Gohari, F. S. Aliee, and H. Haghighi, A new confidence-based recommendation approach: Combining trust and certainty, *Information Sciences*, vol. 422, pp. 21–50, 2018.

[9] Y. Jiang, J. Liu, M. Tang, and X. Liu, An effective web service recommendation method based on personalized collaborative filtering, In: *Proceedings of the 2011 IEEE International Conference on Web Services*, Washington, USA, pp. 211–218, July 4–9, 2011.

[10] C. Kaleli, An entropy-based neighbor selection approach for collaborative filtering, *Knowledge-Based Systems* vol. 56, pp. 273–280, 2014.

[11] H.-J. Kwon, T.-H. Lee, J.-H. Kim, and K.-S. Hong, Improving prediction accuracy using entropy weighting in collaborative filtering, In: *IEEE 2009 Symposia and Workshops on Ubiquitous, Autonomic and Trusted Computing*, Brisbane, Australia, pp. 40–45, July 7–9, 2009.

[12] J. Zhang, Y. Lin, M. Lin, and J. Liu, An effective collaborative filtering algorithm based on user preference clustering, *Applied Intelligence*, vol. 45, no. 2, pp. 230–240, 2016.

[13] Z. Zheng, H. Ma, M. R. Lyu, and I. King, QoS-aware web service recommendation by collaborative filtering, *IEEE Transactions on Services Computing*, vol. 4, no. 2, pp. 140–152, 2011.

[14] J. Bobadilla, A. Hernondo, F. Ortega, and A. Gutiérrez, Collaborative filtering based on significances, *Information Sciences*, vol. 185, no. 1, pp. 1–17, 2012.

[15] P. Melville, R. J. Mooney, and R. Nagarajan, Content-boosted collaborative filtering for improved recommendations, In: *Proceedings of the 18th National Conference on Artificial Intelligence*, Edmonton, Canada, pp. 187–192, July 28–August 1, 2002.

[16] L. Zhang, X. Chen, N.-N. Guan, H. Liu, and J.-Q. Li, A hybrid interpolation weighted collaborative filtering method for anti-cancer drug response prediction, *Frontiers in Pharmacology*, vol. 9, art. no. 1017, 11 pages, 2018.

[17] H. N. Kim, A. Alkhaldi, A. E. Saddik, and G. S. Jo, Collaborative user modeling with user-generated tags for social recommender systems, *Expert Systems with Applications*, vol. 38, no. 7, pp. 8488–8496, 2011.

[18] H. Zhang, D. Yue, C. Dou, K. Li, and X. Xie, Event-triggered multi-agent optimization for two-layered model of hybrid energy system with price bidding based demand response, *IEEE Transactions on Cybernetics*, 2019, DOI: 10.1109/TCYB.2019.2931706.

[19] Z.-H. Zhan, Z.-J. Wang, H. Jin, and J. Zhang, Adaptive distributed differential evolution, *IEEE Transactions on Cybernetics*, 2019, DOI: 10.1109/TCYB.2019.2944873.

[20] W. Liu, Z. Wang, X. Liu, N. Zeng, and D. Bell, A novel particle swarm optimization approach for patient clustering from emergency departments, *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 4, pp. 632–644, 2018.

[21] W. Liu, Z. Wang, Y. Yuan, N. Zeng, K. Hone, and X. Liu, A novel sigmoid-function-based adaptive weighted particle swarm optimizer, *IEEE Transactions on Cybernetics*, vol. 23, no. 4, pp. 632–644, 2019.

[22] J. S. Breese, D. Heckerman, and C. Kadie, Empirical analysis of predictive algorithms for collaborative filtering, In: *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, Madison, USA, pp. 43–52, July 24–26, 1998.

[23] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, Item-based collaborative filtering recommendation algorithms, In: *Proceedings of the 10th International Conference on World Wide Web*, Hong Kong, China, pp. 285–295, May 1–5, 2001.

[24] Y. Shi and R. C. Eberhart, Empirical study of particle swarm optimization, In: *Proceedings of the 1999 IEEE Congress on Evolutionary Computation*, Washington, USA, pp. 1945–1950, July 6–9, 1999.

[25] Y. Shi and R. C. Eberhart, Parameter selection in particle swarm optimization, In: *Proceedings of the 7th International Conference on Evolutionary Programming*, San Diego, USA, pp. 591–600, March 25–27, 1998.

[26] A. Ratnaweera, S. K. Halgamuge, and H. C. Watson, Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients, *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 240–255, 2004.

[27] V. Campuzano, L. Montermini, M. D. Moltò, L. Pianese, M. Cossee, F. Cavalcanti, E. Monros, F. Rodius, F. Duclos, A. Monticelli, F. Zara, J. Cañizares, H. Koutnikova, S. I. Bidichandani, C. Gellera, A. Brice, P. Trouillas, G. De Michele, A. Filla, R. De Frutos, F. Palau, P. I. Patel, S. Di Donato, J.-L. Mandel, S. Cocozza, M. Koenig, and M. Pandolfo, Friedreich's ataxia: Autosomal recessive disease caused by an intronic GAA triplet repeat expansion, *Science*, vol. 271, no. 5254, pp. 1423–1427, 1996.

[28] K. Reetz, I. Dogan, A. S. Costa, M. Dafotakis, K. Fedosov, P. Giunti, M. H. Parkinson, M. G. Sweeney, C. Mariotti, M. Panzeri, L. Nanetti, J. Arpa, I. Sanz-Gallego, A. Durr, P. Charles, S. Boesch, W. Nachbauer, T. Klopstock, I. Karin, C. Depondt, J. M. Vom Hagen, L. Schöls, I. A. Giordano, T. Klockgether, K. Bürk, M. Pandolfo, and J. B. Schulz, Biological and clinical characteristics of the European Friedreich's Ataxia Consortium for Translational Studies (EFACTS) cohort: A cross-sectional analysis of baseline data, *The Lancet Neurology*, vol. 14, no. 2, pp. 174–182, 2015.

[29] K. Reetz, I. Dogan, R.-D. Hilgers, P. Giunti, C. Mariotti, A. Durr, S. Boesch, T. Klopstock, F. J. R. de Rivera, L. Schöls, T. Klockgether, K. Bürk, M. Rai, M. Pandolfo, and J. B. Schulz, Progression characteristics of the European Friedreich's Ataxia Consortium for Translational Studies (EFACTS): A 2 year cohort study, *The Lancet Neurology*, vol. 15, no. 13, pp. 1346–1354, 2016.

[30] W. Yue, Z. Wang, B. Tian, A. Payne, and X. Liu, A collaborative-filtering-based data collection strategy for Friedreich's ataxia, *Cognitive Computation*, vol. 12, pp. 249–260, 2020.

[31] W. Yue, Z. Wang, B. Tian, M. Pook, and X. Liu, A hybrid model- and memory-based collaborative filtering algorithm for baseline data prediction of Friedreich's ataxia patients, *IEEE Transactions on Industrial Informatics*, 2020, DOI:10.1109/TII.2020.2984540.

[32] T. Schmitz-Hübsch, S. T. Du Montcel, L. Baliko, J. Berciano, S. Boesch, C. Depondt, P. Giunti, C. Globas, J. Infante, J.-S. Kang, B. Kremer, C. Mariotti, B. Melegh, M. Pandolfo, M. Rakowicz, P. Ribai, R. Rola, L. Schls, S. Szymanski, B. P. van de Warrenburg, A. Dürr, and T. Klockgether, Scale for the assessment and rating of ataxia: Development of a new clinical scale, *Neurology*, vol. 66, no. 11, pp. 1717–1720, 2006.

[33] B. Qu, N. Li, Y. Liu and F. E. Alsaadi, Estimation for power quality disturbances with multiplicative noises and correlated noises: A recursive estimation approach, *International Journal of Systems Science*, vol. 51, no. 7, pp. 1200–1217, May. 2020.

[34] M. Gao, Z. Zhu and Y. Niu, Robust H_2/H_∞ control for a class of time-varying nonlinear stochastic systems with state- and control-dependent noises, *International Journal of Systems Science*, vol. 51, no. 7, pp. 1218–1228, May. 2020.

[35] D. Zhao, Z. Wang, G. Wei and Q.-L. Han, A dynamic event-triggered approach to observer-based PID security control subject to deception attacks, *Automatica*, vol. 120, Art. no. 109128, Oct. 2020.

[36] Y. Shen, Z. Wang, B. Shen and F. E. Alsaadi, H_∞ state estimation for multi-rate artificial neural networks with integral measurements: A switched system approach, *Information Sciences*, vol. 539, pp. 434–446, Oct. 2020.

[37] M. Wang, Z. Wang, Y. Chen and W. Sheng, Observer-based fuzzy output-feedback control for discrete-time strict-feedback nonlinear systems with stochastic noises, *IEEE Transactions on Cybernetics*, vol. 50, no. 8, pp. 3766–3777, Aug. 2020.

[38] B. Shen, Z. Wang, D. Wang and H. Liu, Distributed state-saturated recursive filtering over sensor networks under Round-Robin protocol, *IEEE Transactions on Cybernetics*, vol. 50, no. 8, pp. 3605–3615, Aug. 2020.

[39] D. Liu, Z. Wang, Y. Liu and F. E. Alsaadi, Extended Kalman filtering subject to random transmission delays: Dealing with packet disorders, *Information Fusion*, vol. 60, pp. 80–86, Aug. 2020.

[40] L. Zou, Z. Wang, J. Hu and Q.-L. Han, Moving horizon estimation meets multi-sensor information fusion: development, opportunities and challenges, *Information Fusion*, vol. 60, pp. 1–10, Aug. 2020.

[41] J. Cao, Z. Bu, G. Gao and H. Tao, Weighted modularity optimization for crisp and fuzzy community detection in large-scale networks, *Physica A: Statistical Mechanics and its Applications*, vol. 462, pp. 386–395, Nov. 2016.

[42] J. Cao, Z. Bu, Y. Wang, H. Yang, J. Jiang and H.-J. Li, Detecting prosumer-community group in smart grids from the multiagent perspective, *IEEE Transactions on Systems Man Cybernetics-Systems*, vol. 49, no. 8, pp. 1652–1664, Aug. 2019.

[43] M. Wang, Z. Wang, Y. Chen and W. Sheng, Event-based adaptive neural tracking control for discrete-time stochastic nonlinear systems: A triggering threshold compensation strategy, *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 6, pp. 1968–1981, Jun. 2020.

[44] Y. Liu, B. Shen and H. Shu, Finite-time resilient H_∞ state estimation for discrete-time delayed neural networks under dynamic event-triggered mechanism, *Neural Networks*, vol. 121, pp. 356–365, Jan. 2020.

[45] X.-M. Zhang, Q.-L. Han, Zidong Wang and B.-L. Zhang, Neuronal state estimation for neural networks with two additive time-varying delay components, *IEEE Transactions on Cybernetics*, Vol. 47, No. 10, pp. 3184-3194, Oct. 2017.

[46] J. Li, Z. Wang, H. Dong and F. Han, Delay-distribution-dependent state estimation for neural networks under stochastic communication protocol with uncertain transition probabilities, *Neural Networks*, vol. 130, pp. 143–151, Oct. 2020.

[47] Y. Liu, Q. Cheng, Y. Gan, Y. Wang, Z. Li and J. Zhao, Multi-objective optimization of energy consumption in crude oil pipeline transportation system operation based on exergy loss analysis, *Neurocomputing*, vol. 332, pp. 100–110, Mar. 2019.

[48] Y. Liu, S. Chen, B. Guan and P. Xu, Layout optimization of large-scale oil-gas gathering system based on combined optimization strategy, *Neurocomputing*, vol. 332, pp. 159–183, Mar. 2019.

Wenbin Yue received his B.Sc. degree in 2013 and M.Sc. degree in 2015, both in information technology from the Queensland University of Technology, Brisbane, Australia. He is currently pursuing the Ph.D. degree in computer science at Brunel University London, London, UK. His research interests include machine learning, recommendation system, big data analysis and bioinformatics.

Zidong Wang (SM'03-F'14) was born in Jiangsu, China, in 1966. He received the B.Sc. degree in mathematics in 1986 from Suzhou University, Suzhou, China, and the M.Sc. degree in applied mathematics in 1990 and the Ph.D. degree in electrical engineering in 1994, both from Nanjing University of Science and Technology, Nanjing, China.

He is currently a Professor of Dynamical Systems and Computing in the Department of Computer Science, Brunel University London, UK. From 1990 to 2002, he held teaching and research appointments in universities in China, Germany and the UK. His research interests include dynamical systems, signal processing, bioinformatics, control theory and applications. He has published 220+ papers in IEEE Transactions and 60+ papers in Automatica. He is a holder of the Alexander von Humboldt Research Fellowship of Germany, the JSPS Research Fellowship of Japan, William Mong Visiting Research Fellowship of Hong Kong.

Prof. Wang serves (or has served) as the Editor-in-Chief for Neurocomputing, the Deputy Editor-in-Chief for International Journal of Systems Science, and an Associate Editor for 12 international journals including IEEE Transactions on Automatic Control, IEEE Transactions on Control System Technology, IEEE Transactions on Neural Networks, IEEE Transactions on Signal Processing, and IEEE Transactions on Systems, Man, and Cybernetics-Systems. He is a Fellow of the IEEE, a Fellow of the Royal Statistical Society and a member of program committee for many international conferences.

Weibo Liu received the B.S. degree in electrical engineering from the Department of Electrical Engineering & Electronics, University of Liverpool, Liverpool, UK, in 2015, and the Ph.D. degree in artificial intelligence in 2019 from the Department of Computer Science, Brunel University London, Uxbridge, UK. His research interests include big data analysis and deep learning techniques. He serves as an Associate Editor for the Journal of Ambient Intelligence and Humanized Computing. He is a very active reviewer for many international journals.

Bo Tian received the B.Eng. degree in automation from Beihang University, Beijing, China, in 2015, and is now pursuing his Ph.D. degree in control science and engineering in Beihang University, Beijing, China. From Mar. 2018 to May 2018, he was a visiting student in the Department of Computer Science, Brunel University London, UK. His research interests include stochastic control and estimation, information theory, and intelligent data analysis. He is an active reviewer for some international journals.

Stanislao Lauria has a Laurea in Physics and Cybernetics from Universita' Federico II, Napoli (Italy) and a PhD in Cybernetics from the University of Reading (UK). He is a Lecturer at Brunel University (UK). He has been research fellow at both the University of Reading and University of Plymouth (UK). He is a computer scientist with an interest in robotics, HCI, robotics and social media, AI, robotics and education, image analysis, data processing. He has contributed to the implementation of several algorithms and architectural models to develop various complex systems and has published several journal and conference papers.

Xiaohui Liu received the B.Eng. degree in computing from Hohai University, Nanjing, China, in 1982 and the Ph.D. degree in computer science from Heriot-Watt University, Edinburg, U.K., in 1988.

He is currently a Professor of Computing at Brunel University. He leads the Intelligent Data Analysis (IDA) Group, performing interdisciplinary research involving artificial intelligence, dynamic systems, image and signal processing, and statistics, particularly for applications in biology, engineering and medicine. Professor Liu serves on editorial boards of four computing journals, founded the biennial international conference series on IDA in 1995, and has given numerous invited talks in bioinformatics, data mining and statistics conferences.