Future Generation Computer Systems 83 (2018) 653–673

Contents lists available at ScienceDirect

Future Generation Computer Systems

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

A hybrid quantum-induced swarm intelligence clustering for the urban trip recommendation in smart city

R. Logesh [a,*], V. Subramaniyaswamy [a], V. Vijayakumar [b], Xiao-Zhi Gao [c], V. Indragandhi [d]

[a] School of Computing, SASTRA University, Thanjavur, Tamilnadu, India
[b] School of Computing Sciences and Engineering, VIT University, Chennai, Tamilnadu, India
[c] Machine Vision and Pattern Recognition Laboratory, School of Engineering Science, Lappeenranta University of Technology, Lappeenranta, Finland
[d] School of Electrical Engineering, VIT University, Vellore, Tamilnadu, India

ARTICLE INFO

Article history:
Received 30 November 2016
Received in revised form 5 July 2017
Accepted 30 August 2017
Available online 18 September 2017

Keywords:
Recommender system
Collaborative filtering
Clustering
Quantum-behaved particle swarm optimization
Swarm intelligence
Bio-inspired computing

ABSTRACT

The development of internet technologies has brought digital services to the hands of common man. In the selection process of relevant digital services to the active target user, recommender systems have proved its efficiency as a successful decision support tool. Among many successful techniques incorporated to generate recommendations, collaborative filtering has been widely used to make similarity-based predictions for the recommendation of the relevant list of items to the users. As an advancement, utilizing clustering mechanisms with collaborative filtering for grouping similar users as clusters can enhance the efficiency of the recommendation generated. Though many clustering mechanisms have been employed to group similar users in the existing works, incorporation of bio-inspired clustering has yet to be explored for the generation of optimal recommendations. In this paper, a novel user clustering approach based on Quantum-behaved Particle Swarm Optimization (QPSO) has been proposed for the collaborative filtering based recommender system. The proposed recommendation approach has been evaluated on real-world large-scale datasets of Yelp and TripAdvisor for hit-rate, precision, recall, f-measure, and accuracy. The obtained results illustrate the advantageous performance of proposed approach over its peer works of recent times. We have also developed a new mobile recommendation framework XplorerVU for the urban trip recommendation in smart cities, to evaluate the proposed recommendation approach and the real-time implementation details of the mobile application in the smart-cities are also presented. The evaluation results prove the usefulness of the generated recommendations and depict the users' satisfaction on the proposed recommendation approach.

1. Introduction

Recommender Systems (RS) predict interesting personalized items for target user among a large collection of available items. To make predictions, the main goal of an RS is to match better items to user's requirements and interests. The general term items can be classified as any user consumables such as food, books, music, movies, e-services, etc. The emerging popularity of e-commerce web portals, e.g. Netflix.com, TripAdvisor.com, Amazon.com, Flipkart.com, etc. - have created massive interests in RS research to provide effective recommendations to end users. The demand of RS has attracted many researchers to the field and progressive research has been conducted in recent years in a wide manner [1–14].

Generally, the RSs are classified based on the techniques used and the major classifications are hybrid, Content-based, Collaborative filtering [15,16]. Among all other techniques, Collaborative Filtering (CF) is widely used with RSs and very successful with its prediction performance [17,18]. In the Collaborative Filtering Recommender System (CFRS), the prediction of items for the target user is made through the assessment of ratings provided by other users for items. Based on the assessment and prediction model, the CFRS can be divided into the item-based and user-based model. As an enhancement to CFRS, clustering techniques were exploited to generate personalized recommendations to meet user's requirements [19–22]. In the user-based CFRS, the clustering technique employs a similarity measure to group the users based on the similar item-ratings given by different users. The items organized based on similarity ratings in a particular cluster of users is recommended to any new user who has similar preferences to the user group. There are many clustering techniques used to generate personalized recommendations in user-based CFRS such as K-means, SOM, fuzzy c-means, etc. But still, bio-inspired clustering

* Corresponding author.
E-mail address: logeshr@outlook.com (R. Logesh).

http://dx.doi.org/10.1016/j.future.2017.08.060
0167-739X/

techniques are not explored for user-based CFRS. This article has attempted to employ a novel bio-inspired swam intelligent model for clustering of users in user-based CFRS, and the obtained results are compared with recommendation models with other existing clustering techniques.

In recent years, research on the development of clustering techniques has attained notable attention in various data mining applications such as web analysis, information retrieval, text mining, pattern recognition, visualization, and image segmentation [23–26]. With the development of real-time clustering based recommender systems, the information processing problems make the recommendation process more complex. The utilization of traditional clustering algorithms such as K-means clustering has some drawbacks in obtaining optimal solutions for large-scale application problems [27–31]. The formal logics-based traditional clustering algorithms fail to help the recommendation algorithms to generate recommendations in fast and efficient manner. To overcome such drawbacks of traditional clustering algorithms, the recent researchers have introduced bio-inspired swarm intelligence for clustering technology. Many researchers have developed many nature-inspired intelligent algorithms to solve real-life engineering problems to generate better optimal solutions [32–39]. Bio-Inspired meta-heuristic approaches are better than traditional models as they are ecologically inspired and their meta-heuristics are specifically designed to tackle complex a large scale problems. Bio-inspired algorithms are very familiar in solving the optimization problems which traditional approaches failed to have an effective or an efficient solution. In many real-world practical problems, the bio-inspired algorithms have proved its performance and recognized as the best solution providing method due to its exploitation of nature-inspired meta-heuristics. The recent trends in the development of hybrid techniques for complex real-world problems, bio-inspired algorithms are highly preferred to solve modern day global optimization problems. Swarm Intelligence inherits the characteristics of biological systems and produces promising results for data analytical models. With the proven efficiency over solving global optimization problems, swarm intelligence models were studied in depth, and the obtained results pave the path for new ideas. As an example, the combinational clustering models of PSO (Particle Swarm Optimization) with C-means and K-means have achieved better clusters over conventional algorithms [40–42].

The new clustering algorithms based on swarm intelligence have a better correlation of data and results with better quality of clusters through attaining great adaptability and rapid convergence. The swarm intelligence based clustering algorithms were adopted by various domains such as neurocomputing, decision-support systems, pattern recognition and big data to experience its enhanced performance [43–53]. With the above line, this article proposes a novel clustering ensemble method with swarm intelligence for recommender systems to solve information overload problem. In this work, we propose a new user clustering approach based on Quantum-behaved Particle Swarm Optimization (QPSO) for better clustering through addressing the drawbacks of traditional clustering based RSs. Quantum-behaved Particle Swarm Optimization (QPSO) is new swarm intelligence algorithm with improved ability over PSO by producing an effective solution for global optimization problems with fewer adjustable parameters [54,55].

The recommendation mechanism of an RS is based on the user's feedback data on the items which are already purchased or consumed. The explicit user ratings are used to generate recommendation list as input along with the implicit user behavior such as user purchase pattern, browsing history, links, etc. The modeling of user preference from their numerical ratings is most commonly used approach. The users' ratings on the 5 point rating scale represent the users' feedback for the particular item as 5 for "most loved" and 1 for "completely hated". The online giants such as Amazon and Netflix analyze the users' rating pattern in their past to predict users' preferences and needs for their upcoming choice of items. After predicting the ratings for non-purchased items, the recommender system organizes the predicted ratings and then ranks accordingly from highly estimated predicted ratings. Finally, the list of top-n items with highly predicted ratings may be recommended to the users. For the enhancement of the recommendations, the traditional recommendation models are combined to frame a hybrid recommendation approaches. Two or more recommendation approaches are combined to form a hybrid recommender system. The hybrid recommender systems have lesser drawbacks over the individual recommendation approaches. In most cases, collaborative filtering based recommendation models are combined with other techniques as a hybrid recommendation approach to overcome ramp-up problem. There are many variants of hybrid recommender systems based on the hybridization method used. The familiar hybridization methods of the recommender systems include weighted, switched, mixed, cascade, meta-level, feature combination and augmentation.

1.1. Need for travel recommender systems in a smart city

In the travel and touristic context, Smartphones help users to overcome navigational difficulties by searching and organizing the required information. In the e-tourism domain, the recent research focuses on mobile technology based recommender systems to provide recommendations on the go. The main aim of the research is to exploit user data and social context to enhance the user acceptance of the generated recommendations. The utilization of mobile technologies for providing travel services benefits both end users and service providers. The development of mobile apps is considered as a significant investment by the travel service providers and helps them to maintain real-time connections with their users. The travel mobile apps are the cost effective solutions than traditional web-based applications by means of enhanced usability, intuitiveness, and attractiveness. Travel destination is the geographical location, which is the important entity of the travel services. With the recent development of Information Communication Technology has changed the prediction mechanism for determining the relevant destination for the active target user. The destination is an area or place that makes a decision to plan a travel trip. In the real-world scenario, a single destination or multiple destinations are determined to generate travel recommendations based on user interests.

With the rapid development and utilization of social networks has helped travel recommender systems to learn the changing user interests for the improvement of the quality of travel related services. The massive user generated data on the online social media is considered to be a significant challenge to grab relevant information for the particular travel. Many existing approaches specifically focus on recommending attractive locations to the users. But still, there is a huge gap yet to be explored for the generation of personalized travel recommendations. Generally, the travel information comprises of heterogeneous data such as images, texts, and videos. For making tailored travel recommendations, the RSs need to be capable of processing the heterogeneous data from social media. The explicit user feedback collection is termed as an active approach, and implicit user feedback collection is known as the passive method for mining the user's interests. For the travel applications, implicit user feedback plays a significant role in predicting relevant-attractive locations for the users. As an effective decision support tool, RSs are widely used as the travel management applications to organize travel and destinations. With employment of hybrid prediction strategies, RSs increases the user sophistication which acquiring travel services. The elimination of manual evaluation of locations reduces judgemental mistakes and increases the number of location choices to be selected. The predictions of travel recommendations are based on the user preferences, profile, interests and relevant activities over geographical distribution of locations with respect to temporal constraints.

The maintenance of the travel recommender systems is a major limitation yet to be addressed. The update of user responses on the locations or point of interests in real-time is also a key factor that affects the quality of recommendations. To address the above-said limitations, the easily available social network data is coupled with the recommender systems. The exploitation of social media data and mobile technology helps end user to update their changing interests and receive recommendations accordingly. In the traditional travel service recommender systems, the recommendation generation is considered as a simple digital service to make recommendations on a map with pin pointers and personalized routes. The existing recommendation approaches in the travel, and e-tourism domain requires specific knowledge base and intelligent technologies to deal with the semantic data and relations between the users and locations. The travel service has to be enhanced as a smart service by extending the regular service with the utilization of digital data generated by other relevant services. The objective of the smart service is to automate the human activities and assist the end users digitally.

Beyond destination determination, personalized trip generation is also a complex task in the travel recommender systems. As the personalized trip consists of one or more POI, modeling the user's preferences based on the available historical data and their online behavior is required to predict the relevant-unvisited POIs. After organizing the relevant POIs, a personalized travel trip has to planned based on the constraints such as distance, travel feasibility, travel mode, etc. There are other constraints such as availability and accessibility of the POI and traveling time needs to meet during planning the trip. The social and economic factors make a huge impact on the user's satisfaction for the generated travel trip. With the increased number of social media users, the impact of the travel recommendations may be immediately reflected in ratings and reviews for the destinations. The reviews of the locations can be analyzed for the polarity, and such detailed analysis helps in filtering POIs before presenting the recommended list to the user.

Travel recommender systems have become popular among decision support tools for making personalized trip recommendations. Travel planning is a significant building block of the sustainable transportation system for an efficient smart city. Travel recommender systems help travelers and commuters by generating personalized trip routes and relevant locations of their interests. The generation of physical world travel recommendation based on the user's preferences and interests on the social network is a complex task. Typically, generating feasible travel plan is a challenging task with the urban destinations or point of interests (POI) due to the interests' match of the traveler and available time span to visit the location. The filtering task based on the ranking of the POIs/destinations along with the available time-sequence of the traveler is a laborious process. The situation becomes more complicated while adding the complex transport networks of the smart cities for the recommendation generation, as the distance between multiple POIs/destination is also a considerable constraint. An interesting aspect is that touristic travelers are interested in spending more time at the locations of their interests rather than transit time between the locations. Here comes the need of an intelligent assistant to support the travelers by recommending interesting destination/POIs based on their social network activities. Along with the location recommendations, travel trip planning is also a complex problem to be addressed. The travel trip planning problem can be solved by matching the users' preferences with the location/POIs features by considering the distance between the POIs, visiting hours, opening hours, transport availability, and traffic feasibility. In this paper, we present a new mobile recommendation framework XplorerVU for the urban trip recommendation in smart cities. The newly developed recommendation framework is used to evaluate the proposed QPSO based CFRS in the smart cities as a real-time scenario. The implementation details of the mobile XplorerVU application in the smart cities Tiruchirapalli (India) and Thanjavur (India) is discussed in detail.

Our main contributions in this paper are as follows.

- We present Quantum-behaved Particle Swarm Optimization based Collaborative Filtering Recommender System for personalized POI recommendation.
- To enhance the recommendation generation process, we extend our proposed QPSO based collaborative recommender system as Quantum Induced Clustering Ensemble (QICE) approach based collaborative recommender system through exploiting multiple swarm intelligent algorithms.
- We propose and develop XplorerVU as a mobile recommendation framework to validate the presented recommendation approaches in the real-time scenarios.
- We have experimentally validated our proposed recommendation approaches on two large-scale datasets of Yelp and TripAdvisor. The obtained results demonstrate the improved performance of proposed approaches over existing approaches by means of efficiency and accuracy.
- We have also validated our recommendation approaches in real-time scenarios at smart cities of Tiruchirapalli (India) and Thanjavur (India).

The remainder of the article is organized as follows: The Section 2 will brief the previous literature on RSs, clustering approaches, and mobile recommendation approaches. Section 3 portrays the Quantum-behaved Particle Swarm Optimization (QPSO), and Section 4 depicts the proposed QPSO based CFRS and the extension of proposed user clustering approach through clustering ensemble method. Then Section 5 provides a detailed explanation on new mobile recommendation framework XplorerVU, and Section 6 briefs the experimental evaluation and depicts the explanations on the results obtained. Finally, Section 7 outlines the conclusions with future work guidelines.

2. Related work

Recommender systems are commonly used decision support tool, used to assist users in generating relevant and interesting recommendations based on the predicted ratings. Goldberg et al. [56] has introduced RS in 1992 and rapidly developed with many approaches to producing precise recommendations through enhanced prediction of ratings. Generally, the rating prediction mechanism is the key indicator to recognize the preference of the user to decide and rank the recommendation of items [57,58]. In recent years, recommender systems have been employed in various domains such as e-commerce, e-services, e-tourism and entertainment applications. Many global technical giants including Netflix, Amazon, Google and Facebook user recommender systems to make an alternate suggestion of products or services to their users. In other words, the most of their business is generated through the employed recommender systems on their portals. As a proven example, the recommender system of Netflix has achieved 75 percent of subscriber watch count including online video streaming and DVDs through mail [59]. Hence, the recommender systems should be efficient enough to satisfy the customer needs to maintain user base and enhance the product sales.

2.1. Recommendation algorithms

The recommender system algorithms were mainly classified into three major categories based on the ratings prediction approaches adapted, and they are collaborative filtering, content-based and hybrid. The collaborative filtering based RS algorithms analyze the ratings matrix for the pattern to make predictions [17,57,60,61]; content-based models generates recommendations based on the comparing the characteristics or features of items with the user profiles; and hybrid approaches merge the collaborative and content-based techniques in various combinations to generate efficient recommendations [62,63]. Among above recommendation methods, collaborative filtering approaches are highly familiar and used widely due to its advantageous domain independency and minimal information requirement for prediction of ratings [64–67]. CF technique computes similarities between users through analyzing the items' ratings and further calculates a weighted average of ratings provided by similar users for the target user to generate recommendation [58].

CF algorithms are generally classified as Memory-Based CF and Model-Based CF. Memory-Based CF RSs generate recommendations based on user or item similarities with the help of entire user–item rating database. Memory-Based CF can be further categorized as user-based CF and item-based CF [15,16]. The user-based CF generates recommendations to the target user, based on similarities between the target user and other users. The user similarities are computed to generate recommendations in user-based CF with an assumption of present similar users with similar taste can have identical taste/ interest in future too. The item-based CF generates recommendations to the target users based on the similarities between different items. The item similarities are computed to generate a recommendation in item-based CF with the assumption that the similar items to the user's purchase history items may have the privilege to make a new purchase in future. The Model-Based CF approaches exploit user–item rating database to address scalability and sparsity problems through working on reduced data in offline mode [16,68].

2.2. User clustering and clustering ensemble

The popular Model-Based CF approaches include clustering [19,21,22] and dimensionality reduction processes [16,58,69]. As CF approaches handle users' personal details and preferences regarding items while construction of personal profiles, it is vulnerable to privacy attacks. Bilge and Polat [20] has presented privacy preserving CF techniques to solve the privacy issues faced by CF techniques through exploiting clustering model. Gupta and Tripathy [70] have presented a new model with Backpropagation Neural Network to train combinational hybrid model of CFRS and Content-Based RS. Based on objective information and subjective opinions from domain experts, fuzzy based RS can strengthen the traditional CFRS and address the cold-start and sparsity problems [71]. In addition, users can present their opinions to the RS with the help of fuzzy linguistic model which can be more useful to incorporate users' online opinion from social media. Birtolo et al. [72] item-based CFRS enhanced with Fuzzy clustering model and generated trust-aware recommendations with clustering based CF. Tsai and Hung [19] introduced Cluster Ensemble (CE) techniques for user-based CFRS to generate personalized recommendations. After the in-depth survey, we have not seen any effective utilization Swarm Intelligence for user-based CFRS.

In recent times the clustering based recommendation approaches have created new paradigm to generate efficient personalized recommendations in various application domains [10,11,73–76]. Various famous clustering algorithms are employed with the recommender systems to generate user clusters based on the similarity between the users. The utilization of traditional clustering algorithms such as K-means clustering has some drawbacks in obtaining optimal solutions for large-scale application problems [27,28]. To overcome such drawbacks of traditional clustering algorithms, the recent researches have introduced bio-inspired swarm intelligence for clustering technology. Swarm Intelligence inherits the characteristics of biological systems and produces promising results for data analytical models. With the proven efficiency over solving global optimization problems, swarm intelligence models were studied in depth, and the obtained results pave the path for new ideas. As example, the combinational clustering models of PSO (Particle Swarm Optimization) with C-means and K-means have achieved better clusters over conventional algorithms [40–42,77–79]. In this paper, we present a new idea of utilizing MWO and PSO with Quantum-behaved Particle Swarm Optimization (QPSO) for clustering through Clustering Ensemble for generating personalized recommendations. Xu and Wunsch [23] have classified clustering into hard and fuzzy types. The hard clustering maps every object to a particular single group whereas the fuzzy type of clustering use membership degree between various groups and different objects for clustering [80]. Bezdek et al. [81] have proposed FCM (Fuzzy c-means), and in the present scenario, it is most popular fuzzy based clustering method. In FCM clustering, the similarity of the elements is taken into consideration along with cluster centers to minimize the criterion function. FCM clustering has higher advantages for the large datasets with elements overlapping in multiple groups. With the need for addressing few shortcomings of FCM, alternative approaches have been explored for fuzzy clustering. Zhang et al. [82] have proposed an extension version FCM with a genetic heuristic search strategy to find the interval weights for the attributes of data to improve the clustering performance.

The successful employment of metaheuristic optimization algorithms such as ACO (Ant Colony Optimization), GA (Genetic Algorithm) and PSO has solved problems like trapping at local minima in optimization problems [83,84]. Due to its simplicity and versatility, PSO has become very popular among metaheuristic algorithms, and it has been used as a significant tool in many applications [85]. With this motivation, many PSO based approaches have been proposed for both hard and fuzzy clustering [85–87]. According to [85], PSO can be used as a standalone model or combinational hybrid model with FCM for clustering methods. Many hybrid clustering models with the help of PSO has proven the improved accuracy over many traditional clustering models such as K-means and FCM [85–87]. However, PSO based models are comparatively slower than conventional clustering models, and tuning of the parameter before discovering the solution is also a notable problem.

There are many modified versions proposed changes on original PSO to attain improved results and performance time. Zhang et al. [88] have proposed a new version of PSO as IDPSO (improved self-adaptive particle swarm optimization) with auto-tuning of parameters to achieve better performance. Izakian and Abraham [83] have developed a hybrid fuzzy clustering model FCM–PSO by combining FPSO and FCM as an extension of FPSO. The experimental evaluation of FCM–PSO has surpassed the results of standalone models of FCM and FPSO. Wilkin and Huang [27] have developed a single iterative clustering algorithm called k-means algorithm, which is very simple to implement on any dataset. K-means algorithm is one of the popular traditional clustering algorithms, suffers from optimal solution problem of local optima. Pei et al. [89] as an extension to the K-means algorithm used PSO with the core idea of k-means to utilize the global optimization capability of swarm intelligence. The combinational model of k-means and PSO has produced significantly better results over k-means algorithm. Similar to PSO, MWO is a new effective meta-heuristic algorithm for global optimization [90]. Due to its simplicity of use algorithm, it can be employed in the different application for better results. The experiments by An et al. [90] has produced relatively better results comparing to GA, biogeography-based optimization, PSO and group search optimization while solving the complex, large-scale optimization problems. The phototaxis and Levy flights of the moths have inspired the researchers to develop new nature-inspired metaheuristic algorithm as Moth Search Algorithm (MSA) [38]. The MSA optimization approach has performed superior over state-of-art metaheuristic optimization algorithms, and it is validated with seven real-world problems. A new PSO variant is presented by combining quantum mechanics with it as QPSO (Quantum-behaved Particle Swarm Optimization) to enhance the global search ability of traditional PSO [36,37,54]. The QPSO variant of traditional PSO is easier to implement as it has only a few parameters to adjust. Fang et al. [91] have proved the iterative equation of QPSO as global convergent, and QPSO has attracted the interests of many researchers various domains. QPSO has better performance over PSO in solving multiple optimization problems through which it can be deployed in various real-time application problems [92,93]. To the best of our knowledge, there is no utilization QPSO for generating optimal solutions in recommender systems domain.

In general, every clustering algorithm is effective and efficient to deal with specific problems, and they have their limitations too. For instance, the results obtained from the clustering algorithms are sensitive to initialization and parameters. Most of the clustering results are found unreliable in determining the actual number of clusters generated. Some clustering algorithms produce variable results with the same dataset. There is no such global clustering algorithm that can produce optimal results with datasets of different types of various structures. To address these problems, Clustering Ensemble (CE) is identified as an effective way [94–97]. CE integrates the clustering results of various clustering models or the same clustering model with different parameters to produce a new clustering result, which is known as a consensus problem solution. CE approach has considerable advantages over single clustering algorithm such as improved quality, categorical attribute based clustering, and effective detection and handling of noises and isolated points. CE has the capability to deal data from distributed sources and able adapt parallel processing for clustering. CE methods can be classified into similarity, transformation, relabeling, and graph-based methods [98]. The proposed RS adapts similarity based CE for better user clustering. Jia et al. [99] have presented a novel selection strategy as SELSCE (Selective Spectral Clustering) to organize a promising committee. The more focus on hard to be classified or misclassified data can produce considerably best results.

2.3. Travel recommender systems

Travel recommender systems have received a notable attention from the researchers in the recent years due to the utilization of heuristic algorithms for recommendation generation [100–106]. The main objective of the trip recommendation generation is to satisfy the user travel requirements. The process of travel recommendation has various constraints such as user preferences, location/POI characteristics, available time space, budget limitations, etc. Considering the personal attitude of the traveler, the influencing parameters for the recommendation generation can be adjusted [107]. Brilhante et al. [108] have presented a web-based travel recommender called TripBuilder through exploiting Flickr and Wikipedia data. Similarly, [109] have presented a genetic algorithm based touristic planner as CT-Planner for travel plan recommendation. Notably, mostly travel/trip recommendation systems lack in the personalization as the recommendation generation is considered the location availability as 24×7. Hasuike et al. [110] have considered the time dependency of the travel between the location and the POI availability times for the travel trip planning. The most of the existing travel recommendation systems are not fully real-time travel assistants as still combinational priority is not given to time and traffic constraints as these are the key factors that affect user satisfaction level in urban travel.

The main contribution of this paper is to introduce novel user clustering based on Quantum-behaved Particle Swarm Optimization (QPSO) for in CFRS for enhanced recommendations. We have introduced a hybridized CE embracing Swarm Intelligence and Quantum mechanics to benefit RS engine in real-time scenarios. The proposed QPSO based CFRS is evaluated on two real-time large-scale datasets of Yelp and TripAdvisor as standalone approach and hybridized CE approach. As a necessitated initiative on clustering based recommender systems, we have validated our proposed model for the recommendation of the trip in two smart cities in India through our presented mobile recommendation framework XplorerVU.

3. Quantum-behaved particle swarm optimization

We present the preliminaries required for the understanding the proposed recommendation approach. The technological advancements of the internet and web technologies had created an enormous amount of data which creates information overload problem. The processing of the massive online data requires new hybrid techniques help recommender systems to generate better relevant items. The complicated biological systems inspire the researchers to solve complex user clustering problems of the recommender systems. The main goal of the user clustering process of the proposed recommendation approach is to enhance the recommendation generation process. In this section, we describe the Quantum-behaved Particle Swarm Optimization based on the preliminary information relevant to PSO. Inspired from the trajectory analysis of PSO and the quantum mechanics, the mean of previous best positions are utilized to frame QPSO algorithmic model. QPSO has a major difference from the PSO, as it does not require velocity vector from the particles and it has only one parameter to adjust during the implementation process. With the ease of implementation capability, in the recent time, many researchers have been attracted to QPSO development and adaptation for various application domains.

3.1. PSO algorithm

Several scientists have developed an optimization technique inspired by the schools of fish and flocks of birds as PSO algorithm. From the initial development of PSO in 1995, there are many variations has been coined for the better optimal solution for various application domains. In the traditional PSO algorithm, on the dimensional space *DIM*, each of *max* individuals is treated as a particle without volume. For the particle *par*, position and velocity vectors in the kth iteration can be represented as follows:

$$POS_{par}(k) = \left[pos_{par,1}(k), pos_{par,2}(k), \ldots, pos_{par,DIM}(k)\right] \quad (1)$$

$$VEL_{par}(k) = \left[vel_{par,1}(k), vel_{par,2}(k), \ldots, vel_{par,DIM}(k)\right] \quad (2)$$

Based on the above-given position and velocity vectors, the particle moves according to the following equations:

$$\begin{aligned} vel_{par,par\_var}(k+1) &= \left[inewei \cdot vel_{par,par\_var}(k)\right] \\ &+ coeff\,1 \cdot RAN_{par,par\_var}(k) \cdot \left[pbest_{par,par\_var}(k)\right. \\ &\left. - pos_{par,par\_var}(k)\right] + coeff\,2 \cdot ran_{par,par\_var}(k) \\ &\cdot \left[gbest_{par\_var}(k) - pos_{par,par\_var}(k)\right] \end{aligned} \quad (3)$$

$$pos_{par,par\_var}(k+1) = pos_{par,par\_var}(k) + vel_{par,par\_var}(k+1) \quad (4)$$

In the above equations, $par = 1,2,\ldots,DIM$ and $par\_var = 1,2,\ldots,DIM$. *coeff1* and *coeff2* are the acceleration coefficients, and *inewei* is the inertia weight. The best previous position with the best value from the objective function is called as the personal best (*pbest*) position, and it is defined as, $pbest_{par}(k) = [pbest_{par,1}(k), pbest_{par,2}(k), \ldots, pbest_{par,DIM}(k)]$. The position of the best particle among all particle in the dimensional space *DIM* is known as the global best (*gbest*) position and it is defined as, $gbest(k) = [gbest_1(k), gbest_2(k), \ldots, gbest_{DIM}(k)]$. By considering minimization problems, the $pbest_{par}(k)$ can be updated without losing the generality by,

$$pbest_{par}(k) = \begin{cases} pos_{par}(k) \\ \quad if\ fun\left[pos_{par}(k)\right] < fun\left[pbest_{par}(k-1)\right] \\ pbest_{par}(k-1) \\ \quad if\ fun\left[pos_{par}(k)\right] \geq fun\left[pbest_{par}(k-1)\right], \end{cases} \tag{5}$$

Here, $fun(\cdot)$ is the objective function.

As the result, *gbest(k)* can be determined through following equation:

$$gbest(k) = pbest_{glob}(k) \tag{6}$$

where, $glob = \arg\min_{1\leq par\leq DIM}\left\{fun\left[pbest_{par}(k)\right]\right\}$.

The two parameters $RAN_{par,par\_var}(k)$ and $ran_{par,par\_var}(k)$ are the uniformly distributed random numbers within the range [0,1]. Hence the random number parameters can be represented as $RAN_{par,par\_var}(k)$, $ran_{par,par\_var}(k) \sim U(0,1)$. The value of $vel_{par,par\_var}(k)$ is generally restricted within the range interval [-vel_{max}, vel_{max}].

3.2. QPSO algorithm

The major limitation of the traditional PSO is its inability to guarantee global convergence, as it is prone to fall into local optima beyond its faster-converging capability. Motivated by the quantum mechanics and trajectory analysis of PSO, QPSO (Quantum-behaved Particle Swarm Optimization) has been developed for the optimization problem. The trajectory analysis depicts that, PSO algorithm may achieve convergence, while every particle converges to the local attractor $pla_{par}(k)$ and it can be defined as follows:

$$pla_{par,par\_var}(k) = \varphi_{par,par\_var}(k)\cdot pbest_{par,par\_var}(k) + \left[1-\varphi_{par,par\_var}(k)\right]\cdot gbest_{par\_var}(k) \tag{7}$$

In the above equation, $par\_var = 1,2,\ldots,DIM$ and $\varphi_{par,par\_var}(k)$ is defined as follows:

$$\varphi_{par,par\_var}(k) = coeff1\cdot RAN_{par,par\_var}(k)/\left[coeff1\cdot RAN_{par,par\_var}(k) + coeff2\cdot ran_{par,par\_var}(k)\right] \tag{8}$$

Here, $RAN_{par,par\_var}(k)$ and $ran_{par,par\_var}(k)$ are the uniformly distributed random numbers within the range [0, 1]. As the acceleration coefficients, *coeff1* and *coeff2* are assigned to be equal (i.e. $coeff1 = coeff2$), the $\varphi_{par\_var}(k)$ sequence is distributed uniformly as a random number within the range (0, 1). It can also be defined as $\varphi_{par\_var} \sim U(0,1)$.

In QPSO, it is presumed that each particle is spin-less and embraces quantum behavior. As a result, the particle's state is written off as Ψ and $|\Psi|^2$ is used as the probability density function of the particle's position. It is assumed that particle *par* moves in the dimensional space *DIM* with the potential δ at the $pla_{par,par\_var}(k)$ in the *par_var*th dimension ($1 \leq par\_var \leq DIM$) to guarantee the convergence. The following iterative equation can be obtained by solving the Schrödinger equation through exploiting Monte Carlo method, to update the particle's position.

$$\begin{cases} pos_{par,par\_var}(k+1) = pla_{par,par\_var}(k) \\ \quad + CE\_coeff\cdot\left|C_{par\_var}(k) - pos_{par,par\_var}(k)\right|\cdot\ln(1/rn1) \\ \quad if\ rn2 \geq 0.5 \\ pos_{par,par\_var}(k+1) = pla_{par,par\_var}(k) \\ \quad - CE\_coeff\cdot\left|C_{par\_var}(k) - pos_{par,par\_var}(k)\right|\cdot\ln(1/rn1) \\ \quad if\ rn2 < 0.5, \end{cases} \tag{9}$$

Here, *rn1* and *rn2* are the different random numbers generated by the uniform probability distribution within the range (0, 1) and mean best position is the mean of the *pbest* positions of all the particles is known as $C(k)$, and it is defined as follows:

$$C_{par\_var}(k) = (1/DIM)\sum_{par=1}^{DIM} pbest_{par,par\_var}(k) \tag{10}$$

When the value of generated random number *rn2* is greater than or equal to 0.5, then the former formula is used to update the particle's position, else the particle's position is updated according to the second one. The *CE_coeff* is the Contraction–Expansion coefficient parameter, and it is used to control the convergence rate of the particle. Fixed and varying are two simple methods adopted to tune the *CE_coeff* parameter to optimize the performance of the algorithm during the search. QPSO performs better when *CE_coeff* <1.781 and with varying tuning method if $CE\_coeff_1 < CE\_coeff_2$, then QPSO delivers good performance.

4. Proposed quantum-behaved particle swarm optimization based collaborative filtering recommender system

We develop QPSO based CFRS for the generation of personalized POI recommendation. With the aim of achieving efficient recommendations, the QPSO algorithm is exploited for the user clustering process. As a complete recommendation approach, our proposed QPSO based CFRS utilizes the LBSN data which contains user's check-in data and user ratings for the prediction of relevant locations based on the user clusters. Our proposed QPSO based CFRS comprises of three segments namely, clustering, prediction, and recommendation. The organization of the proposed RS is depicted in Fig. 1. In the first segment, the users in the dataset are clustered through QPSO algorithm. In the second segment, the neighbor search for the active target user is made to map the user with appropriately matching user cluster. Finally, based the neighbor and user's current cluster the ratings are predicted, and the top-n list of most relevant items are generated to the user. In Fig. 1, at the first step of the proposed QPSO based CFRS, the datasets are divided into 80% as training set and remaining 20% is employed as a testing set. In the next step, users are clustered using QPSO algorithm to produce user clusters by minimizing the dissimilarities between them. Each clustering job is executed individually as an iterative process with ranging cluster numbers to generate better cluster to produce recommendations with higher accuracy rate. To reduce the convergence time of QPSO, we employ K-Means Operator (KMO) for the enhanced clustering process [54]. The generated user clusters are used in the neighborhood search for the active target user and after determining the matched cluster, the active target user is added to the matching user cluster. Then, Based on the ratings of the user cluster, the ratings are predicted for the target user. Finally, predicted ratings are ranked and transformed into top-n recommendations.

The prediction process of QPSO based collaborative recommender system determines the neighbors for the target users by adding the user to the user cluster of higher similarity. During the neighbor determination process, the similarity computation process extracts the correlation information between the users. As

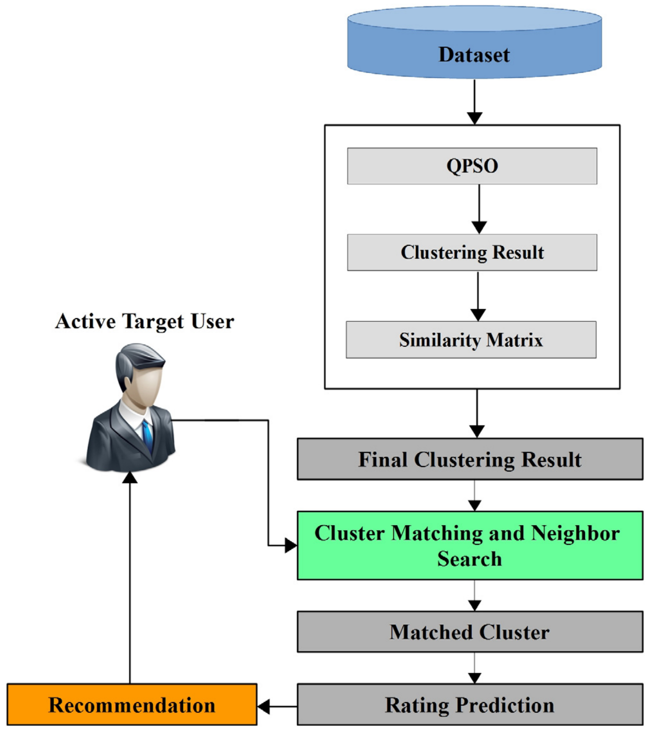

Fig. 1. Proposed QPSO based collaborative filtering recommender system.

a contribution to the prediction process, the most similar users to the active target user are selected as neighbors. Generally, users with pre-defined similarity weights are also chosen as neighbors. After selection of neighbors to the active target users, as a next step, the ratings of items for the target user is predicted based on the ratings provided by the neighbors. The rating prediction process adopts the averaging and PCC measures to compute similarity based ratings for relevant items.

The recommendation section of the proposed RS manages to generate a highly relevant list of items as recommendations to the active target user. An appropriate list of items is generated as recommendations based on the user's neighbors in the clusters. Generally, in the recommended list of items, the highly rated items of the highly similar users were highly ranked and given higher preferences. Finally, the recommendation list is sorted out based on the threshold limit of the active target user. For the organization of the recommendation list of POIs, the threshold limit is set by users as their preferences. The personalized organization of the recommending the items to the target user have higher acceptance rate with enhanced satisfaction levels.

4.1. Enhanced user clustering through clustering ensemble

The proposed QPSO based CFRS approach is extended as Quantum Induced Clustering Ensemble (QICE) approach. The main aim of designing QICE is to utilize the enhanced performance of the clustering ensemble approaches to achieve effective user cluster groups. The organization of the proposed QICE based RS is depicted in Fig. 2. The main difference between the QPSO based CFRS and QICE is that the former one generates the user clusters with the single QPSO algorithm and the later one employs multiple bio-inspired optimization algorithms along with QPSO algorithm for user clustering process. Similar to the QPSO based CFRS recommendation approach the QICE approach also comprised of three major segments for predicting and recommending the relevant POIs to the target user. In the first segment, the users in the dataset are clustered through similarity based clustering ensemble model which includes the bio-inspired clustering methods. The statistical ensemble clustering model aggregates the generated user clusters of QPSO, K-PSO, and K-MWO algorithms. Among various swarm intelligent optimization techniques, we have chosen the K-PSO and K-MWO algorithms to generate user clusters along with the QPSO, as the both algorithms are already hybrid in nature with the combination of traditional K-Means algorithm. The K-PSO and K-MWO algorithms have better adaptability with any clustering algorithms and help the clustering ensemble to perform superior over standalone approaches. In the second segment, the neighbor search for the active target user is made to map the user with appropriately mating user cluster. Finally, based the neighbor and user's current cluster the ratings are predicted, and the top-n list of most relevant items are generated to the user.

In Fig. 2, at the first step of the proposed Quantum Induced Ensemble (QICE) approach for CFRS, the datasets are divided into 80% as training set and remaining 20% is employed as a testing set. In the next step, users are clustered using bio-inspired optimization algorithms are executed in parallel to produce user clusters by minimizing the dissimilarities between them. To describe the clustering process of users, the three bio-inspired clustering methods namely, QPSO, K-PSO, and K-MWO are used. As the outcome of the user clustering process, the dissimilarity elements in the same cluster are minimized while grouping the users together. Each clustering algorithm is executed individually and simultaneously to generate clusters with a similarity matrix. Later, based on the similarity matrix generated by the bio-inspired clustering algorithms, a consensus similarity matrix is computed to make clustering ensemble. In the generation of consensus similarity matrix, the average matrix of all the similarity matrices is calculated. It is to be noted that the average matrix differs from the similarity matrix, as the average matrix values range from 0 to 1 and the similarity matrix is always binary in nature. To make the average matrix into binary, a threshold value may be determined to generate definite binary consensus similarity matrix. The generated binary similarity matrix is then transformed into final set clustered users.

Each clustering job is executed individually as an iterative process with ranging cluster numbers to generate better cluster to produce recommendations with higher accuracy rate. Later the generated clustering results are converted into similarity matrix. As the next step, the individual outcome of bio-inspired clustering algorithms (i.e.) similarity matrices is passed as input to the statistical ensemble algorithm. As the outcome of the statistical ensemble process, single consensus similarity matrix is formed, and later it is transformed as an ideal user clusters. These user clusters are used in the neighborhood search for the active target user and after determining the matched cluster, the active target user is added to the matching user cluster. Then, Based on the ratings of the user cluster, the ratings are predicted for the target user. Finally, predicted ratings are ranked and transformed into top-n recommendations. For the organization of the final recommendation list, the threshold limit is determined by users as their preference. The personalized organization of the recommending the items help in achieving higher acceptance rate for the recommendation list with enhanced satisfaction levels.

4.1.1. Combinational approach of k-means with PSO as k-PSO

The combination model of conventional K-means with PSO for the clustering as K-PSO adapts the global optimization ability of PSO and address the local optima problem efficiently. Each particle in K-PSO is a possible clustering solutions comprising of K classes and they are represented as $X_j = (x_1, x_2, \ldots, x_k)$. For all y_m,$m \in N_K$, N_K is the coordinate vector of the center of mth class in the jth particle of the swarm. The actual present situation determines the dimension of the x_j. The optimal solution generated in the solution space is the final clustering result obtained.

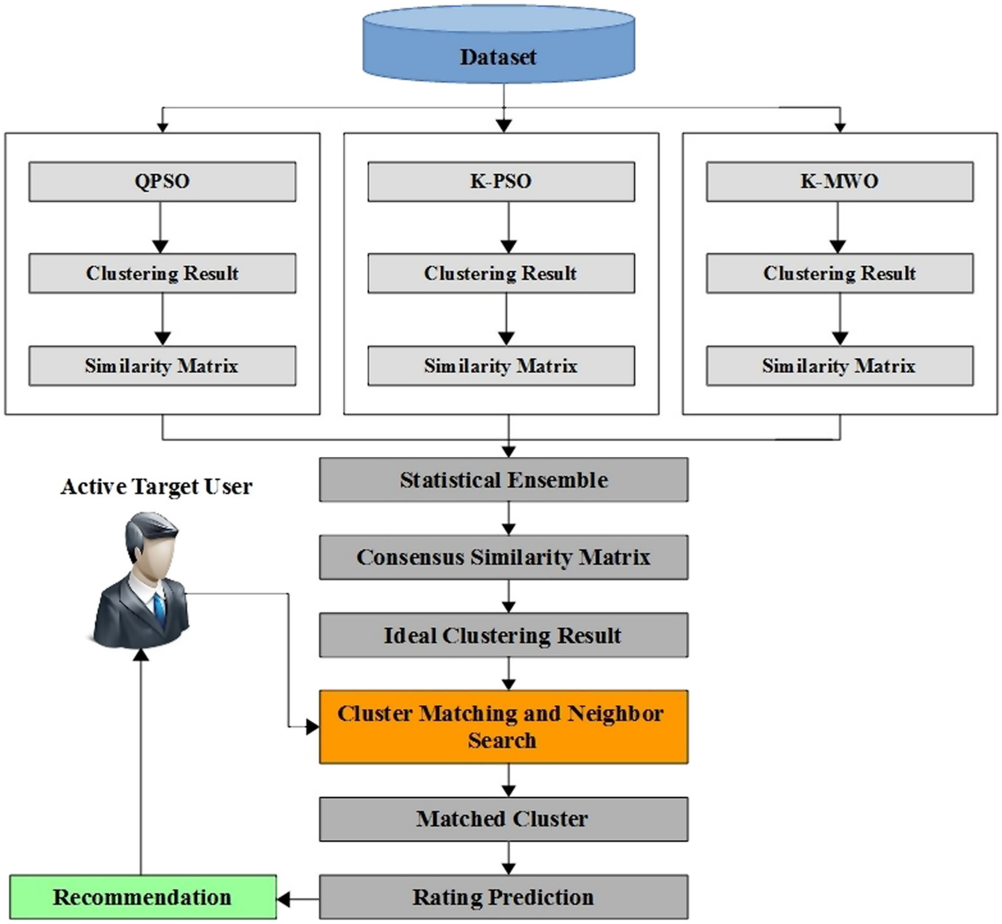

Fig. 2. Proposed quantum induced clustering ensemble based collaborative filtering recommender system.

4.1.2. Combinational approach of k-means with MWO as k-MWO

Mussels Wandering Optimization (MWO) is swarm intelligence based new meta-heuristic method inspired by the locomotion behavior of mussels during the bed formation pattern in their home environment. MWO algorithm is a bio-inspired optimization algorithm, which formulates the evolutionary mechanism mathematically to distribute the mussels' bed patterns by Levy walk and a stochastic decision. A new approach of MWO solves the complex optimization problems in a better manner and performs well compared to other optimization algorithms. A hybrid combinational model of MWO with the conventional K-Means for the clustering of users is proposed.

In K-MWO algorithm, as a first step initializes all N mussels and as a next step, an objective function is used to evaluate the fitness of every mussel by a squared sum error. Based on the calculated fitness value, the top $p\%$ of mussels with their updated position coordinates of next generation are sorted out. Tracing the mussels with higher fitness value helps the learning process to provide better directions on the evolution process. A levy walk within the range of zero to one is computed to determine the displacement of mussels for the updating process. It is to be noted that the updating process of mussels' position is achieved dimension by dimension. As a restriction condition, the new position of the mussels should be out of the limits to avoid the unsuitable field.

The objective function to calculate the fitness of every mussel is as follows.

$$mussel\_fitness = \sum_{j=1 w_j \in cen_m}^{N} \sum_{m=1}^{cen} \left(w_j - cen_m\right)^2 \quad (11)$$

Here, $W_j, j \in N_N$, is the jth data point of the dataset and N is the total number of data points available in the each class, CEN_m, $m \in N_{CEN} = \{1, 2, 3, \ldots, CEN\}$ is the center of mth class, and the number is classes is represented by CEN.

The mussels' position is updated based on its levy walk. The levy walk for every mussel can be calculated using following equation.

$$levy_j = \delta[1 - ran]^{-1/(\varphi-1)} \quad (12)$$

Here, δ represents the scale factor of the walk, ran is a random sample value of the uniform distribution with the range zero to one and φ is the shape parameter ranging $1.0 < \varphi < 3.0$.

Based on the calculated levy walk, the mussel's position is updated by,

$$w_j^{update} = w_j + levy_j \Delta_g \quad (13)$$

Here, Δ_g is the distance between the top p% mussels' w_g and jth mussel and Δ_g can be determined by $\Delta_g = w_g - w_j$. After updating of mussels, the new fitness value is computed, and the new top $p\%$ of mussels are updated along with new w_g.

The MWO algorithm is executed as an iterative process until reaching termination criterion to produce best optimal clustering results. If the termination criterion has not been reached, the levy walk is calculated for updated position of mussels iteratively to produce optimal results.

4.1.3. Clustering ensemble

The proposed QICE based CFRS exploits similarity based clustering approach for user clustering to predict recommendations. The basic clustering results of the bio-inspired clustering algorithms are obtained in the form of $q \times q$ similarity matrix *SIM*. The size of the dataset is generally, q. The obtained individual similarity matrices are co-ordinated as an ensemble to form consensus similarity matrix. Later, the generated consensus similarity matrix is converted into final clustering output. Based on the basic clustering results obtained from the algorithms, it can be classified as soft clustering and crisp clustering. When every data point on the cluster belongs to a single class, then it is called as crisp clustering. The similarity matrix is a binary matrix representing the co-occurrence of data point pairs in the same cluster. After generation of consensus similarity matrix, an average of all similarity matrices is calculated. It is to be noted that the average similarity matrix differs from the basic similarity matrix as its value ranges between zero to one and it is not a binary matrix. For better understanding, when the *SIM* (x, y) is near to one, then the data points x and y tend to belong to same class. A threshold is defined to make average similarity matrix into a binary matrix (e.g. 0.5). If the *SIM* (x, y) is above the threshold value, then the value is assigned as 1 or else 0. Based on this calculation process, definite binary consensus similarity matrix is generated and finally transformed into an optimal clustering result [111]. The proposed QICE based CFRS adapts the clustering ensemble approach from [99,112] for organization of the final user clusters.

Algorithm: Clustering Ensemble of QICE approach

Input: *Basic Cluster from algorithms* (*BC*)
Output: *Clustering Ensemble* (*CE*)

for algo=1, . . . , k
  $CE^{(algo)} = BC^{(algo)}(Dataset)$;
  Generate $q \times q$, $SIM^{(algo)}$ *with respect to* $CE^{(algo)}$;
End *for*
$SIM = \frac{1}{k}\sum_{algo=1}^{k} SIM^{(k)}$;
$CE=BC(SIM)$;
return CE;

In above algorithm, the basic clusters of multiple bio-inspired clustering algorithms are obtained as input and further processed for generation of base clusters. Then, similarity matrix $SIM^{(algo)}$ is generated based on base clusters $CE^{(k)}$. Later an average similarity matrix is computed to form consensus similarity matrix. Finally, based on the consensus similarity matrix, final ensemble clustering is formed, and optimal clustering results are returned.

After determining clusters, the neighborhood is discovered based on the similarity between the users. Computed similarity value extracts the correlation information of the users. Hence highly similar users were chosen as the contributing neighbors in the recommendation prediction process. Here the users with the higher similarity weights influence the threshold value and selected as direct neighbors for the target user. As the next step, the ratings are predicted for the items similar to the rated items of the neighbors. The averaging and PCC measures are used in our proposed QICE based CFRS for the ratings prediction, and final list of Top-N relevant items are generated and recommended to the user.

4.2. User based collaborative filtering

The RSs working with User Based Collaborative Filtering (UBCF) generates personalized recommendations through exploiting the user–item rating matrix, which is generally represented as $usr \times itm$ matrix. In the user–item rating matrix, the *usr* denotes the users with preferences and *itm* represents the items on the system. When the user requests a recommendation from the RS, then the user is called to be a target user and the based on similarity, the neighbor users are computed. Based on the neighbor users' previous ratings on the items, the prediction has been made for the specific item for the target user. In other words, the estimation of recommendations for the target user is made through considering the preferences of similar users with identical tastes [16,113,114]. Table 1 depicts an example user–item rating matrix.

Table 1
A sample user–item rating matrix.

	Item 1	Item 2	Item 3	Item 4	Item 5
User 1	4	3	–	5	2
User 2	2	4	5	–	3
User 3	–	5	1	3	2
User 4	5	3	–	2	–

Generally, there are two ways to determine neighbor users like clustering algorithms and traditional similarity measures. The similarity computation model to determine similitude between existing users and target user plays a vital role in the ratings prediction method of an RS. In the existing literature there are many familiar similarity measures [15,16,113–118] and based on the empirical analyses with respect to CFRS, PCC (Pearson Correlation Coefficient) performs better than other available measures while computing comparisons between users [16,113]. The similarity computation between two users x and y by PCC can be defined as follows:

$$sim\left(x,y\right) = \frac{\sum_{i\in I}\left(Ratings_{x,i}-\overline{Ratings_x}\right)\left(Ratings_{y,i}-\overline{Ratings_y}\right)}{\sqrt{\sum_{i\in I}\left(Ratings_{x,i}-\overline{Ratings_x}\right)^2}\sqrt{\sum_{i\in I}\left(Ratings_{y,i}-\overline{Ratings_y}\right)^2}} \quad (14)$$

Where, $I = \{i_1, i_2, i_3, \ldots, i_n\}$ which represents the set of items, $Ratings_{x,i}$ denotes the ratings of user x for item i, $Ratings_{y,i}$ represents the ratings of user y for item i, $\overline{Ratings_x}$ and $\overline{Ratings_y}$ is the average rating of users x and y respectively.

Similarly, the ultimate goal of a clustering algorithm is to organize similar users into groups as clusters. In the clustering based models, the neighbor users are selected for the target users from the cluster in which they belong to. This study exploits the clustering-based methods similar to neighbor users defining models used in the literature [19–22,119,120,121]. It is also proved that the clustering based models outperform the similarity measures based models while determining the similarity based neighbors for the target user [120]. The PCC measure can also be used to rating prediction process in addition to defining the neighbors for target users [15,16]. The prediction based on PCC is computed as a weighted average of neighbors with respect to the rating of items i and can be defined as follows:

$$pred\left(x,i\right) = \overline{Ratings_x} + \frac{\sum_{y\in NN} sim\left(x,y\right) * \left(Ratings_{y,i}-\overline{Ratings_y}\right)}{\sum_{y\in NN} sim\left(x,y\right)} \quad (15)$$

Where NN represents the nearest neighbor, $Ratings_{y,i}$ is the ratings of user y for item i, $\overline{Ratings_x}$ and $\overline{Ratings_y}$ is the average rating of users x and y respectively.

The utilization of PCC helps in the selection of more similar users per cluster than selecting all *nu* users in a cluster, where *nu* is the total number of users in that cluster. The Maximizing Average Satisfaction is another popular ratings prediction approach which is used to estimate the average of all ratings if items i with a set of neighbor users *nu* [120]. Based on the estimated prediction, the list of top rated items will be recommended to the target user. Ratings prediction based on Maximizing Average Satisfaction approach is defined as follows:

$$pred\,(x, i) = \frac{1}{nu} \sum_{c=1}^{nu} Ratings_{ci} \tag{16}$$

We have adopted two major prediction approaches Pearson and Maximizing average satisfaction to explore the personalized prediction capabilities of our proposed QPSO based CFRS, and QICE approaches. The utilization of multiple prediction approaches helps to reveal the performance of the user clustering based recommendation models with respect to different approaches. Beyond Pearson and Average prediction approaches, the current research induced to standardize the agent-based and contextual based models for better ratings prediction to generate personalized recommendations.

5. XplorerVU-proposed mobile recommendation framework

In this section, we present XplorerVU, a new mobile recommendation framework as a decision support tool for personalized travel recommendation. We have developed a new mobile recommendation framework to evaluate our proposed recommendation approaches in the real-time smart city scenarios. The mobile recommendation applications are proven as a best decision support tools for the travel applications. Our proposed XplorerVU generates personalized travel recommendations based on the user's personal activities for the urban POIs. The proposed system collects data from various sources in real-time for the generation of tailor made travel recommendations. XplorerVU travel recommendation application has been made available to the end-users as the mobile application on Android phones. As the travel domain is completely different from other recommendation domains, to reduce sparsity, we exploit user's social network profile to analyze the current context and previous check-ins. The significant advantage of XplorerVU is its ability to adapt any contextual changes to provide efficient travel recommendation. Fig. 3 depicts the overview of the XplorerVU architecture. The XplorerVU mobile recommendation framework exploits the user's personal data and preferences from the online social network (OSN) or Location Based Social Network (LBSN). The target user interacts with the XplorerVU travel recommendation engine with the help of XplorerVU smartphone interface. The XplorerVU travel recommendation engine employs the proposed QPSO based CFRS and QICE approaches for the recommendation generation process. After generating the relevant list of POIs to the target user, the XplorerVU travel recommendation engine exploits the inputs from the weather forecast web services and travel planning services to filter POIs for the enhancement of recommendation list. Later the final list of recommendations is presented to the target user through XplorerVU smartphone user interface.

The XplorerVU travel recommendation application is designed to be user-friendly, and it provides an interactive interface to manage the trip constraints for better trip planning. As the advanced provision, users of XplorerVU can customize their preferences in a better way in between the travel, and it immediately reflects with the travel trip. Users can also add their personal constraints for the travel plan such as weather, energy level, budget, and interest categories. To manage the complexities of the travel recommendations, we have combined multiple recommendation approaches to make a hybrid recommendation engine. We have utilized our proposed user clustering approach of Quantum Induced Clustering Ensemble for CFRS. As an outcome, the end users receive organized list of top-n relevant locations. Among the generated top-n locations, users can select list of their choice for more personalization. Before making recommendations for the active target user, the locations are pre-filtered based on the user's preferences and the locations that do not match the user's preferences are not considered for the recommendation process. XplorerVU also avoids recommending already visited places by the users in the recommendation list as those places are considered to be undesirable.

After getting all relevant information required for travel planning and trip recommendation, XplorerVU generates the lists of relevant locations that satisfy the user's requirements. Then immediately, the locations/POIs are presented to the user to through the client interface for obtaining their choice of selection. After receiving the user's choice on the recommended list of locations, the road plans for the trips are generated, and it is presented to the user as the map view. The trip management facility in the XplorerVU helps users in the urban transport to choose better roads for faster transit. The real-time streaming of data helps the system to cover all the updates from weather to sudden incidents that affect the travel. With this functionality, the proposed travel recommendation system can make an interrupt in-between the travel to make an alert on the criticality which helps users to stop the travel and return to their hotel or place of stay.

To describe the filtering process of available locations, we initially filter the locations based on users' interesting categories and weather constraints. Based on the current weather condition match and travel feasibility, the outdoor POIs and locations are not included in the selection list. Suppose, when the user has not selected any interesting category in their preferences, then the locations are chosen accordingly based on the popularity and the ratings provided by the similar users. As the next aspect, the travel mode availability between multiple locations and POIs for the organization of trip with minimum travel distance. For this process, the interesting locations and the POIs for the active target user are clustered into geographical groups. For the users with higher energy level, location within walking distances are given priorities, and for the users with the lower energy levels, the location with the walking mode is discarded in the recommendation list. Though we do not focus on the clustering of the locations, we just cluster locations to enhance the travel activity of the users to have satisfied travel trip.

The optimal selection of the next travel location from the current location the user should satisfy the user's constraints on the budget and time availability. Selection of the next target location may be between various location clusters, and there is no rule to have all the locations within a single location cluster. Along with distance between the locations, the similarity between the user and the location, popularity of the location and the similar user's feedback on the location is also to be considered before making a choice. The XplorerVU raises a force flag to represent an inappropriate choice made by the user to report better available options rather than the selected one. Generally, the locations are sorted based on the global user ratings and active target user's preferences, and the selection is made from the top by satisfying budget constraints. After selecting the location from the list, before making next choice, the availability of the time and budget is updated for better organization of travel. The location selection process gets terminated once the maximum time window for the day gets ended.

After organizing the top-n list of most relevant locations/POIs, the traveling routes between them are analyzed. Typically, the initial routes are not the best routes to match the user needs. As the next stage, the routes are needed to be modified according to the opening time or available time of the POI/location. We use Google Maps API to present our trip plan route to the user as a recommendation. The generated recommendation is subject to be modified based on user's knowledge on the travel plan or other contextual influences. As the final recommendation, XplorerVU recommends optimal travel route with the maximum match to the user's preferences and other constraints. During day travel, the

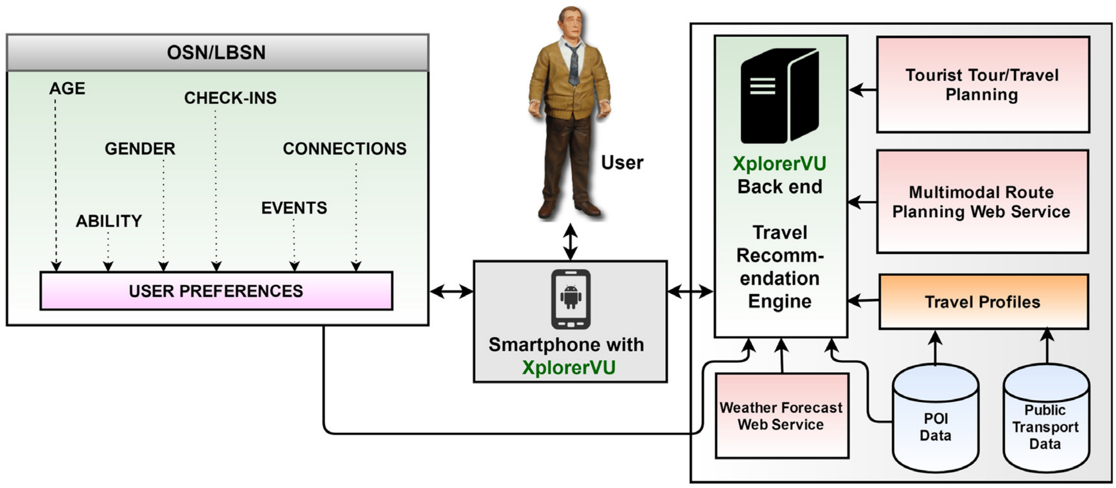

Fig. 3. System overview of XplorerVU mobile recommendation framework.

transportation time between two locations of the recommended trip plan may exceed the actual time considered for the planning. Based on the observations made during the travel, an automated optimization process suggests an alternate route matching the time-window to cover all the locations recommended for that day. Adaptation of the changing context is the significant feature of a good recommender system, and XplorerVU incorporates the changing dynamics of environmental impacts and user preferences. To be specific in the travel and tourism domain, this feature is very important as the user may require changing the plan or adding a new location during travel. As the travelers from different countries have an impact on various cultural heritage values during travel, the proposed XplorerVU is flexible enough to be compatible to assist users in their travel. As XplorerVU utilizes the GPS for the location based services, we have taken primitive steps to reduce the energy consumption of the mobile device through utilizing modern web technologies and connectivity protocols.

Basically, the traditional recommendation applications have no importance to the lunch breaks as they only consider the temporal features of the locations alone. In XplorerVU, we have incorporated a choice of having a lunch break or rest break in-between the travel. Without making any disturbance or distraction on the planned travel route, XplorerVU recommends restaurants and hotels on the travel plan route. Recommendation on the restaurants and hotels are mostly matching to the user's budget class. When the user accepts the recommendation on the restaurant, immediately it is updated with travel plan for the day. The updating of the plan may derive a new plan if the new location is available to be added that matches user's new time-window. The availability of the new matching location will be notified to the user, and the only user has the authority to add the new location or to discard it. The weather forecast details are updated to the users from Yahoo! Weather and weather criticality is used as decisive logic to continue the travel.

Fig. 4 depicts the user interface of XplorerVU mobile application and its recommendations in the smart city—Thanjavur (India). The recommendation generation process of the proposed XplorerVU mobile application presented in Fig. 4 is explained as follows: Our XplorerVU mobile recommender system helps target user in planning a trip and discovering new places. To plan a trip, our XplorerVU recommendation framework collects few inputs from the user such as trip origin location, number of travel days, interest categories, budget constraints and other activities constraints. Based on the user inputs and the user's online activities, the relevant list of POIs are presented to the user. Then, based on the user's selection on the recommended list of POIs, the selected POIs are presented on a map view with the personalized travel plan. Regarding the discovery of new relevant places for the active target user, the only input required is the user's current location. Based on the user's current location, the relevant list of locations is presented to the user in a map view. The XplorerVU helps the users to save the recommendations, travel plans and discovered plans for the future use.

6. Experimental evaluation and discussions

In this section, the proposed QPSO and QICE based recommendation approaches are experimentally evaluated for the analysis of effectiveness, performance, efficiency, and accuracy based on user clustering generated. Experiments were conducted on a PC running on 64-bit Windows 7 operating system with Intel Core i7-5500U clocked at 3.00 GHz and 16 GB of memory. The XplorerVU mobile recommendation framework is evaluated on a smartphone with Snapdragon 410 Quad-Core clocked at 450 MHz with 2 GB of memory. The obtained results of QPSO and QICE are compared with existing approaches such as k-Means, c-Means, PSO, MWO and MSA in the Clustering Based RS scenarios. For the better understanding of the readers, the results and analyses are presented neatly.

6.1. Datasets

The proposed QPSO and QICE based collaborative recommender system are evaluated on two large-scale real-time datasets of Yelp and TripAdvisor. Yelp is a famous location reviewing website, and it acts as an absolute source to make our experimental evaluation. The pre-processing of the dataset removes the users with fewer ratings on venues and results with 39 104 venues, 20 166 unique users, and overall 586 274 ratings. The TripAdvisor is the travel recommendation website comprises of reviews and feedbacks of the locations. Similar to the pre-processing of the Yelp dataset, TripAdvisor dataset is also pre-processed, and users with fewer ratings are removed. The filtered TripAdvisor dataset includes 9149 venues, 13 410 unique users, and overall 152 721 ratings. The statistical comparisons of both datasets are portrayed in Table 2. The datasets are divided into two parts for training and testing as 80 and 20 percent respectively.

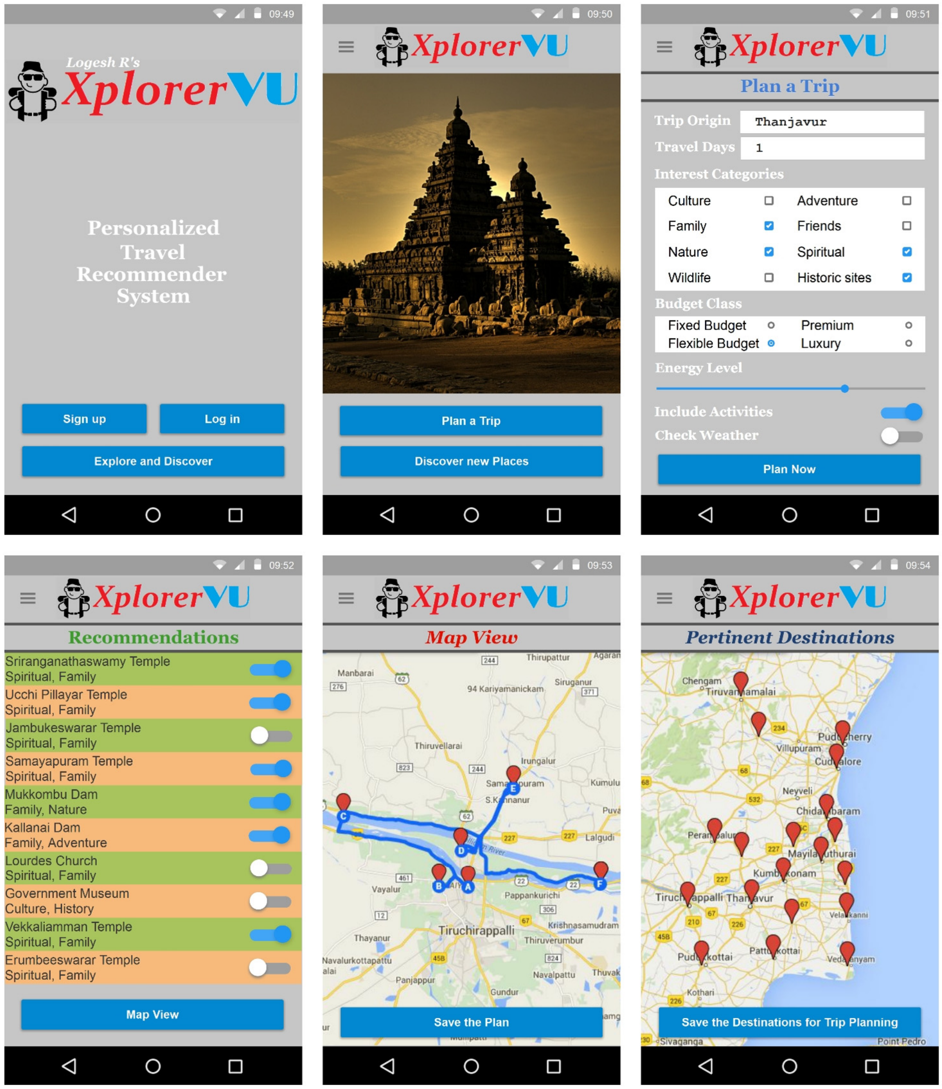

Fig. 4. User interface of XplorerVU with its recommendations in the smart city—Thanjavur (India).

Table 2
Statistical comparison of Yelp and TripAdvisor datasets.

Statistics	Yelp		TripAdvisor	
	User	POI	User	POI
Maximum number of ratings	1234	1189	96	708
Average number of ratings	29.1	15.0	11.4	16.7

6.2. Evaluation metrics

The main aim of the conducted experiments is to evaluate the performance of the proposed QPSO and QICE based collaborative filtering recommender system for its recommendation ability. Experiments are conducted on both Yelp and TripAdvisor datasets. We use four evaluation metrics precision, recall, f-measure, accuracy and hit-rate to evaluate the performance of recommendation approaches.

(A) Hit-rate

Hit-rate is used to evaluate the satisfaction of the user with respect to generated recommendations list. Hit-rate represents the fraction of hits in the recommended list of items which contains the user's interested items. The hit-rate generally computed using the following definition.

$$Hit - rate = \frac{Number\ of\ Hits}{n} \tag{17}$$

Here, *Number of Hits* represents the total number of hits by the user and the total times of recommendation are denoted by *n*.

(B) Precision

The commonly known positive predictive value is also known as precision. Precision is the percentage of recommended items relevant to the user, and it is defined as follows:

$$Precision = \frac{|Reco\_Item\,(user) \cap Relevant\_Item\,(user)|}{|Reco\_Item\,(user)|} \tag{18}$$

Here, the *user* represents the target user in the test data, *Reco_item(user)* is the list of recommended items and *Relevant_item(user)* is the list of items pertinent to the target user in the test set.

(C) Recall

The percentage relevant items that are recommended is known as recall. The recall is also known as sensitivity, and it is defined as follows:

$$Recall = \frac{|Reco\_Item\,(user) \cap Relevant\_Item\,(user)|}{|Relevant\_Item\,(user)|} \tag{19}$$

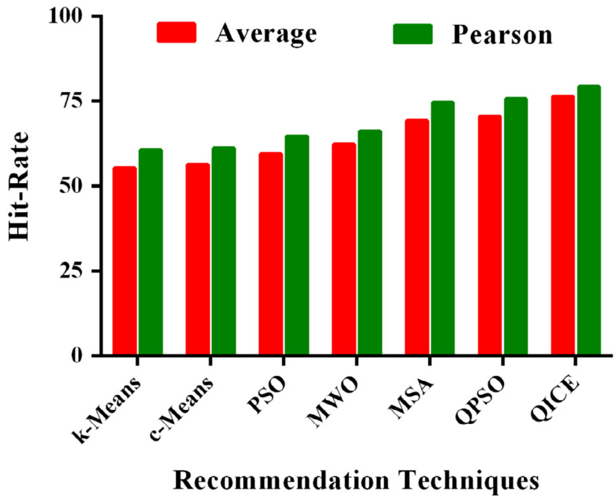

Fig. 5. Comparison of hit-rate on Yelp dataset.

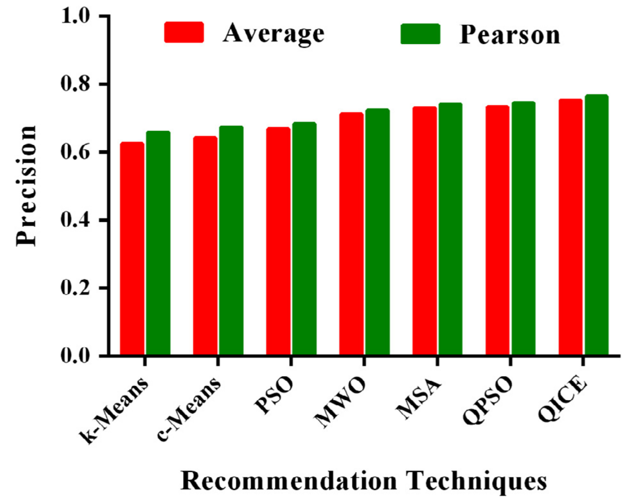

Fig. 7. Comparison of precision on Yelp dataset.

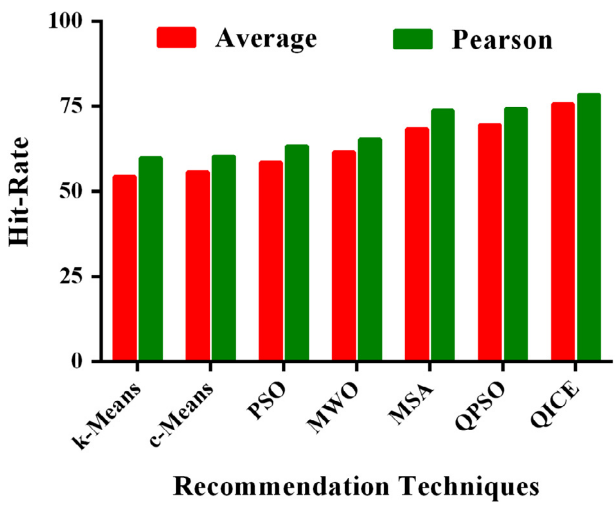

Fig. 6. Comparison of hit-rate on TripAdvisor dataset.

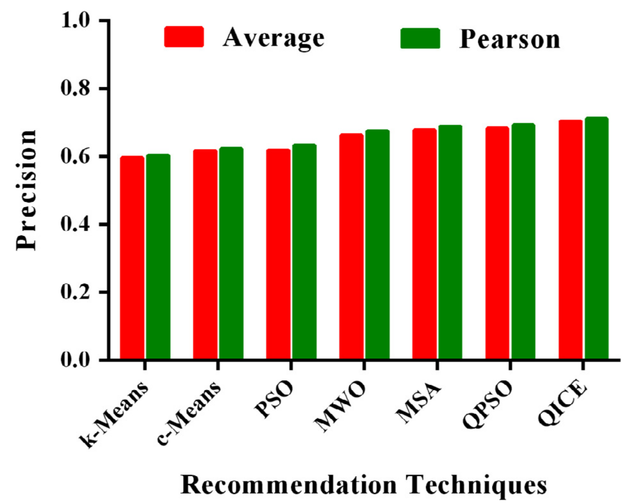

Fig. 8. Comparison of precision on TripAdvisor dataset.

Here, *user* represents the target user in the test data, *Reco_Item(user)* is the list of recommended items and *Relevant_Item(user)* is the list of items pertinent to the target user in the test set.

(D) F-measure

The f-measure metric is the harmonic mean of recall and precision computed and is defined as:

$$F - Measure = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (20)$$

(E) Accuracy

Accuracy metric is a decision support metric, which is based on the selection made with highly matching items from the set of all available items. The binary operation of the prediction process estimates the matching scores of the item to the active target user as "good" or "bad". The accuracy rate is generally computed from the obtained recommendation using the following definition:

$$Accuracy = \frac{\sum TruePositive + \sum TrueNegative}{\sum TruePositive + \sum FalsePositive + \sum FalseNegative + \sum TrueNegative} \quad (21)$$

6.3. Comparison of different recommendation models and discussions

The experiments of the proposed QPSO and QICE based collaborative recommender system are conducted on two large-scale real-time datasets, and the obtained results were compared with existing standalone clustering based recommendation approaches for evaluation approaches. The experiments are conducted with different clustering methods with the user clusters numbers ranging from 3 to 15. The comparisons and the analysis of the experimental results are made with the k-Means, c-Means, PSO, MWO, and MSA in clustering Based CFRS. Figs. 5–16 depicts the obtained results from various recommendation approaches, and Table 3 tabulates the precision, recall, and F-measure achieved for different recommendation approaches with respect to Yelp and TripAdvisor datasets. The obtained results depict the performance of proposed QPSO, and QICE based collaborative filtering recommendation approaches over other recommendation methods. This means that the proposed QPSO and Quantum Induced Clustering Ensemble models for user clustering are capable of providing better recommendations compared to other stand alone clustering models for user clustering. Followed by QICE, the QPSO has comparative performances with QICE, and it can be easily inferred from the results. The MSA based recommendation approach has performed

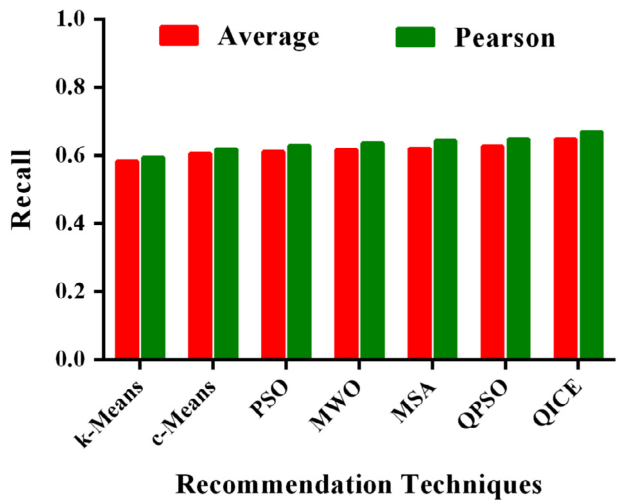

Fig. 9. Comparison of recall on Yelp dataset.

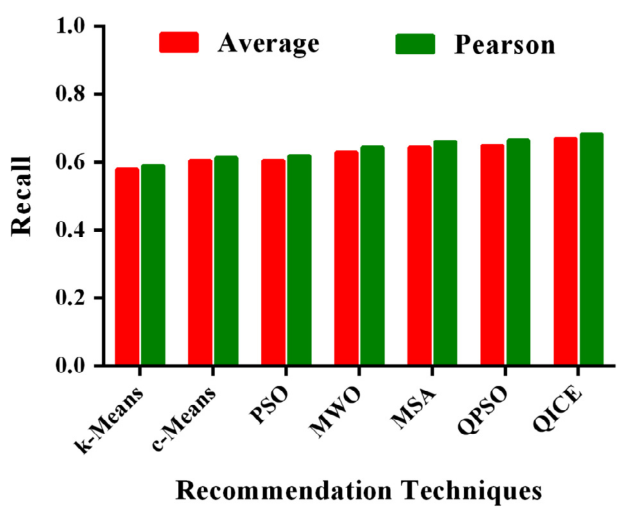

Fig. 10. Comparison of recall on TripAdvisor dataset.

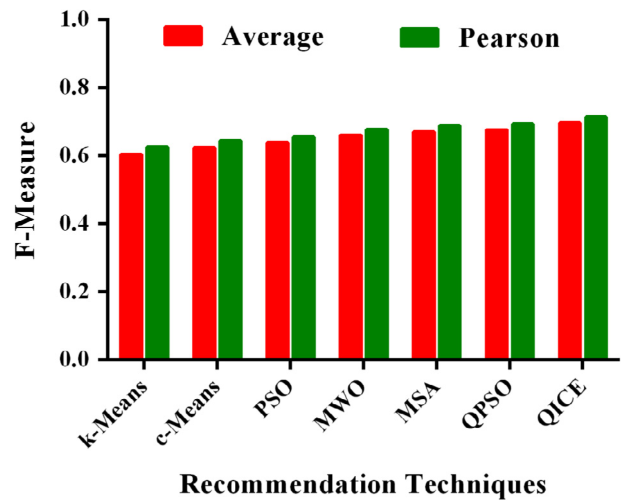

Fig. 11. Comparison of F-measure on Yelp dataset.

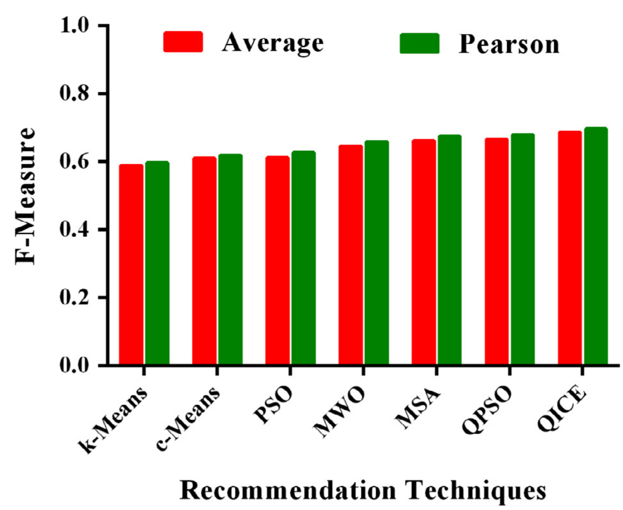

Fig. 12. Comparison of F-measure on TripAdvisor dataset.

comparatively equivalent to the QPSO approach. PSO and MWO based user clustering methods have also performed better, but it cannot outperform the results of standalone MSA, QPSO models and QICE model. While describing regarding the performance of k-Means and c-Means based user clustering approaches, it is to be noted that the recommendations' effectiveness lags compared to other bio-inspired metaheuristic clustering approaches. The higher pertinence rate is achieved by the generated recommendations of QICE based CFRS, as the approach incorporates the meta-heuristic clustering ensemble method for the user clustering and this approach provides the efficient neighborhood selection for user clustering.

We have also evaluated the recommendation accuracy of the recommendations with respect to the number of items recommended ranging from one to ten. We have inferred a notable pattern from the results obtained from both QPSO and QICE approaches as the recommendation accuracy increases with the increase of the number of items to be recommended. This work intentionally exploits the bio-inspired meta-heuristic algorithms to utilize the performance of hybrid clustering approaches through clustering ensemble method to attain better recommendations. User clustering operations are performed independently on both large-scale datasets by proposed recommendation approach. From

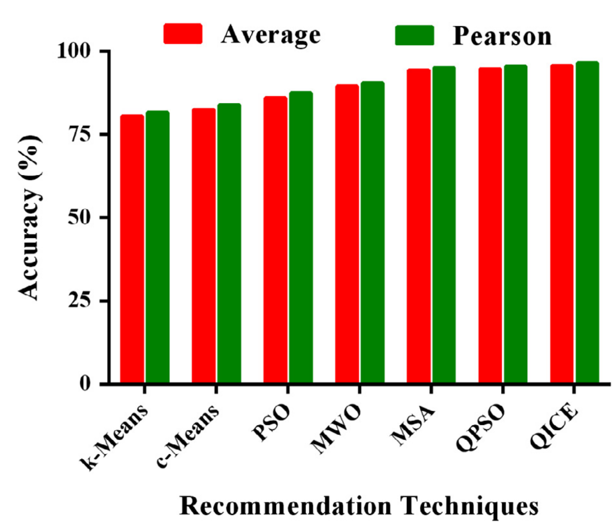

Fig. 13. Comparison of accuracy on Yelp dataset.

Table 3
Precision, recall and F-measure of different recommendation approaches.

Recommendation approach	Prediction type	Yelp			TripAdvisor		
		Precision	Recall	F-measure	Precision	Recall	F-measure
k-Means	Average	0.6255	0.5834	0.6037	0.5967	0.5808	0.5886
	Pearson	0.6598	0.5962	0.6264	0.6046	0.5915	0.5980
c-Means	Average	0.6429	0.6074	0.6246	0.6174	0.6052	0.6112
	Pearson	0.6736	0.6188	0.6450	0.6237	0.6136	0.6186
PSO	Average	0.6692	0.6126	0.6397	0.6194	0.6056	0.6124
	Pearson	0.6846	0.6299	0.6561	0.6341	0.6198	0.6269
MWO	Average	0.7125	0.6175	0.6616	0.6642	0.6287	0.6460
	Pearson	0.7246	0.6374	0.6782	0.6753	0.6439	0.6592
MSA	Average	0.7301	0.6214	0.6714	0.6798	0.6445	0.6617
	Pearson	0.7412	0.6435	0.6889	0.6894	0.6614	0.6751
QPSO	Average	0.7334	0.6267	0.6759	0.6837	0.6496	0.6662
	Pearson	0.7458	0.6485	0.6938	0.6936	0.6658	0.6794
QICE	Average	0.7527	0.6487	0.6968	0.7042	0.6696	0.6865
	Pearson	0.7653	0.6692	0.7140	0.7124	0.6832	0.6975

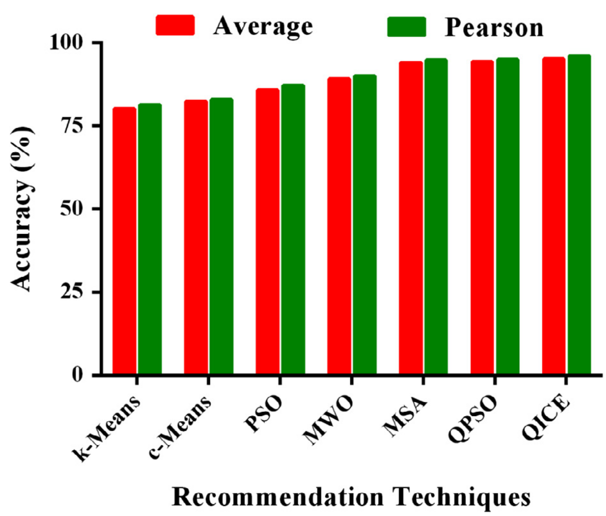

Fig. 14. Comparison of accuracy on TripAdvisor dataset.

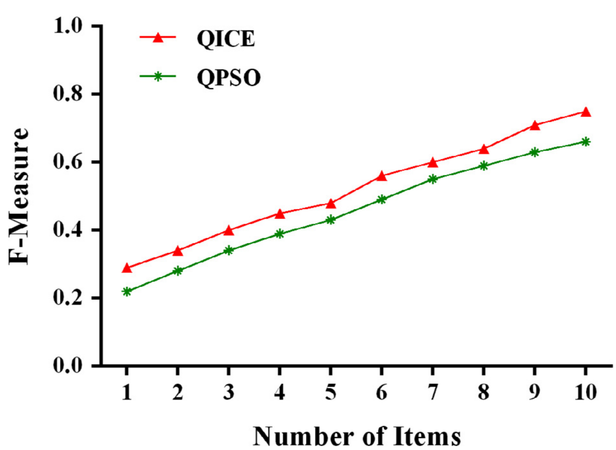

Fig. 15. F-measure with respect to number of items recommended on Yelp dataset.

obtained results, an inference is made that the PCC prediction model generates better results compared to the average prediction model on both datasets. Table 4 portrays the comparison of accuracy and hit-rate of different recommendation approaches.

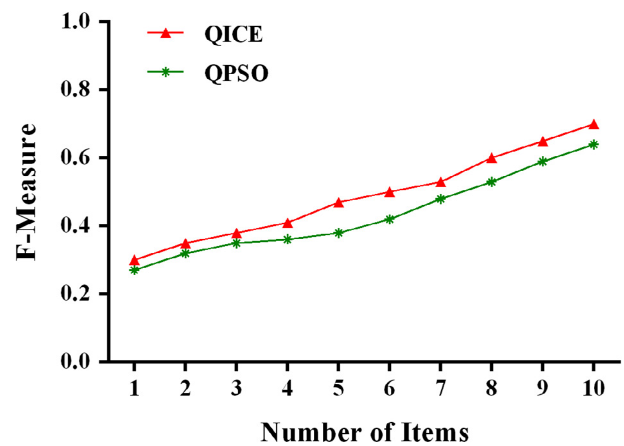

Fig. 16. F-measure with respect to number of items recommended on TripAdvisor dataset.

From the experimental results, the proposed QPSO and QICE based collaborative filtering recommendation approaches perform better over other recommendation methods through accuracy and hit-rate values. The MSA based recommendation approach has also performed well and produced comparable recommendations equivalent to the QPSO approach. PSO and MWO based user clustering methods have also performed better, but it cannot outperform the results of standalone MSA, QPSO models and QICE model. The recommendation performance of k-Means and c-Means based user clustering approaches lags compared to other bio-inspired metaheuristic clustering approaches in both accuracy and hit-rate. Table 5 reveals the minimum and maximum accuracy of the different recommendation approaches along with its corresponding cluster numbers. The recommendation approaches with total 3 user clusters has produced maximum accuracy and the recommendation approaches with total 15 clusters has produced minimum accuracy.

Based on the obtained experimental results, QICE approach has been chosen for the real-time evaluation. The QICE approach is employed on the presented XplorerVU mobile recommendation framework for the evaluation of its recommendation ability. The XplorerVU recommender system is evaluated with overall 92 participants from the smart cities of Tiruchirapalli and Thanjavur for travel recommendation and trip planning during October 2016. The main aim of this evaluation study is to estimate the adaptability of presented mobile recommendation framework and analyze

Table 4
Accuracy and hit-rate of different recommendation approaches.

Recommendation approach	Prediction type	Yelp		TripAdvisor	
		Accuracy	hit-rate	Accuracy	hit-rate
k-Means	Average	80.65	55.34	80.21	54.42
	Pearson	81.74	60.72	81.39	59.86
c-Means	Average	82.61	56.49	82.37	55.67
	Pearson	83.86	61.32	83.12	60.45
PSO	Average	86.12	59.63	85.98	58.58
	Pearson	87.63	64.74	87.29	63.36
MWO	Average	89.66	62.48	89.24	61.65
	Pearson	90.52	66.33	90.13	65.46
MSA	Average	94.43	69.35	94.04	68.43
	Pearson	95.09	74.69	94.86	73.87
QPSO	Average	94.82	70.64	94.36	69.59
	Pearson	95.50	75.74	95.04	74.37
QICE	Average	95.74	76.45	95.24	75.68
	Pearson	96.54	79.36	96.12	78.49

Table 5
Maximum and minimum accuracy of different recommendation approaches.

Recommendation approach	Prediction type	Maximum		Minimum	
		Accuracy	Cluster no.	Accuracy	Cluster no.
k-Means	Average	80.96	3	80.11	15
	Pearson	81.84	3	81.09	15
c-Means	Average	82.92	3	82.07	15
	Pearson	83.97	3	83.14	15
PSO	Average	86.73	3	85.38	15
	Pearson	87.89	3	87.06	15
MWO	Average	89.94	3	89.13	15
	Pearson	90.87	3	90.15	15
MSA	Average	94.88	3	94.02	15
	Pearson	95.79	3	94.75	15
QPSO	Average	94.93	3	94.14	15
	Pearson	95.78	3	94.89	15
QICE	Average	95.91	3	95.12	15
	Pearson	96.95	3	96.09	15

the recommendations from the QICE approach on the real-time scenarios. The usability of the XplorerVU and the pertinence of the generated recommendations are the key areas of focus during the evaluation phase. The users of the XplorerVU were requested to experience and explore the application throughout for in-depth usage, and we have received feedback from them. The user feedback includes various methods such as ratings, short interview, live observations, emotional reactions, etc. Our presented approach is also evaluated for its design and human–computer interaction abilities on ISO 9241-11-110 standards. The evaluation of the XplorerVU on ISO 9241-11-110 standards reveals its effectiveness and efficiency in self-descriptiveness, sustainability, suitability to user requirements, individualization, error tolerance and controllability in a satisfying manner.

The experiments are conducted with the aim to evaluate the proposed recommendation framework for the efficiency, accuracy, and novelty. Generally, the efficiency of the system is classified into two types as computational space and computational time. The effectiveness of the proposed recommender system is determined based on the processing time taken to generate personalized recommendations with improved accuracy. Efficiency can also be defined as the combinational feature that describes both quality of recommendations and time taken to generate recommendations. To reduce the time consumption for the recommendation generation, both online and offline data processing approaches are adopted based on the requirements. The diversified learning mechanism of the proposed recommendation framework helps the system to generate better recommendation with the minimal average processing time. The experimental results reveal the potential capabilities of the proposed travel recommender system with the generation of improved recommendations based on user clustering. Beyond the popularity of the locations, our proposed RS is designed to consider new and novel POIs to be recommended to the active target user. The users specially treat the surprising and novel recommendations, and this feature makes the proposed RS different from existing traditional approaches. The efficient prediction of relevant POIs improves the user experience and makes the RS smart.

The mobile interface of the proposed RS presents the tailored travel recommendations to the user. The map feature of the mobile application is very much interactive and obtains the user inputs for the further personalization. The “save the plan” option helps the user to save the recommended trip for the future visit and the saved trip is considered as the explicit preferences for the further predictions. For the new users, we have designed the recommendation framework to organize the popular POIs within the user’s travel threshold limit to avoid the cold-start issue. The XplorerVU is intelligent enough to leverage user’s implicit and explicit preferences to organize predicted POIs into a personalized trip as per user’s convenience. With the integration of information from the different sources helps the developed recommender system to learn the travel requirements, current contextual attributes, travel motivations and geographical information to make efficient predictions. The automated user clustering process of the proposed recommendation approach by QPSO and QICE determines the similarity between users to enhance the recommendation generation process. The filtering of POIs for the active target user also considers the activities available at the POIs or nearby area to add advantageous travel options for the recommendations.

The mobile interface of the presented XplorerVU recommender system assists users to plan travel trips and tourist activities with a user-friendly planning process. As the XplorerVU is employed on Smartphone of the end user, the current location of the user is always registered with the recommender system with the help of GPS technology. The location-aware ability of the RSs helps in determining the suitable locations that meet the taste of the user. The final recommendation list provided to the user comprises of diversified and relevant recommendations that meet the requirements of the user. To describe the usability of the XplorerVU, the travel recommendation application is user-friendly and adaptable to all users from the common man to experts.

The evaluation XplorerVU is done by users, by making travel plan request with varying constraints of travel. The users are requested to merge their social network profile for better recommendations. The look-and-feel of the XplorerVU has attracted users, and its responsiveness for the query is well received by them. The users' feedback for the generated recommendations and the ratings for the generated travel plan is positive, which depicts the quality of the QICE recommendation approach and XplorerVU's user interface. The QICE based recommendation approach very well matches the user preferences, and the attributes of the locations/POIs considered for the recommendations. Most participant users have shared their credits to the map view facility which is the extension the generated recommendations. The experimental evaluation and the user experience evaluation have clearly portrayed the enhanced performance of the proposed QICE recommendation approach with CFRS through XplorerVU mobile recommendation framework. The improved performance of the QICE based CFRS with XplorerVU mobile recommendation framework has achieved in a reasonable runtime. We have also executed the proposed recommendation approach on high-performance computer clusters to reduce runtime on generating recommendations for real-world applications. It is to be noted that the computational complexity and the quality of the recommendations are major factors in the development of real-time RSs. As the final line, we have presented a better model with efficient computation performance requirements and improve prediction model for user-acceptable recommendations.

6.4. Travel planning as a case study in smart city scenario

An effective travel recommender system not only recommends destinations and relevant POIs to the user but also combines multiple POIs of the recommendations list as a possible travel trip. Travel trip generation problem in tourism domain is commonly known as Tourist Trip Design Problem (TTRP). When the active target user needs to travel from the current location to the destination location by visiting different relevant POIs in a direct route, the RSs should be capable of designing the travel route with the minimal travel time covering the maximum relevant POIs before reaching the destination. In the smart city scenario, the relevant POIs are discovered based on the user preferences initially, and then the travel trip route is designed by combining the relevant POIs in a sequence. Organizing the trips with the mobile application is advantageous as the user may call for a modification in the recommended trip and make amendments. The temporal constraints of the user are considered with more care and weather forecasts are suggested to the users before making final trip plans. Usually, when the system has its interface on the Smartphone, the user behavior is learned for the further recommendation process. With the help of GPS, the travel route of the user is taken into account to validate the recommended travel plan. From the time spent at the POI and the rating score given for that POI, the proposed RS infers the value of the POI category and stores it as implicit preferences for future utilization. When the user spends more time at POI beyond the scheduled time, then the system notifies the user for the extra time consumption at the location. This feature helps the user to reschedule the trip or cover the other visits quicker to reach the destination in planned time.

In the typical smart city scenario, the user may plan the travel trip for multiple days to cover the interesting tourist locations. Hence, the start and end location are same such as stay place or hotel or guest house. The best option available to make an optimal travel plan is to design round trips by starting and ending the travel plan at the user's stay location. Our XplorerVU supports the planning round-trips and discovers the nearby interesting POIs and schedules the travel plan accordingly. For the round-trip generation, XplorerVU considers the temporal constraints of both user and POI to plan a perfect trip and present the trip with the determined duration of the trip with the appropriate time splits for all POI of the trip. When the user has stay in the same city for the multiple days, the interesting sights are distributed to all days based on the location categories to make the trips of all days equally interesting and surprising. The pre-planned trip has an option to be modified or canceled at any time. When the user alters the plan, the modifications are given higher priority, and the changes are flagged as critical preferences for the user. In some cases, the user may modify the plan with the intention of avoiding the breaks planned for food at restaurants, and we have designed our RSs to neglect these minor preferences. When the user continuously avoids the food breaks, and when it crosses the threshold limit, then our system notifies the user to update the preferences for food breaks. In most cases, the presented XplorerVU has recommended satisfying round-trips to the users in the smart cities of Tiruchirapalli and Thanjavur.

In real-time, tourists choose POIs based on their categories to make check-in. The existing approaches only multiple similar POIs to frame a travel trip. This is not treated as a good approach and only similar POIs cannot satisfy a user on the travel. For an example, a trip with multiple restaurants as recommendations can never give satisfaction to the user and for the full day trip recommendations may contain maximum three restaurants. The repetitive presence of the POIs with the same category affects the attractiveness of the recommended trip. The prediction process of the relevant POIs is different from the trip planning procedure, and hence the predicted ratings score for the POI differs from the overall score of the travel trip. The diversity and novelty of the POIs in the travel trip increases the user satisfaction and achieves better feedback. In special cases, some users choose similar POIs for the trip generation. For example, a traveler is on pilgrimage needs to visit temples chooses the list of temples to generate the trips. Our RS is intelligent enough to learn the changing user preferences from single POI to multiple POIs selection for trip planning. From the learned preferences, heuristics are formed to compute the better trip plans to achieve better user satisfaction.

For any mobile application, user interaction and user interface design are the key factors that influence the user's perception on the presented recommendations. A powerful recommendation set presented in a poor way will make very little impact on the end user's interests. Similarly, complex interactions for the recommendation generation also make the negative impact on the RS. Taking these points into consideration, our XplorerVU is designed in an adaptable way to meet the changing demands of the users in real-time. XplorerVU can recommend a single destination, multiple POIs, and complete travel plans. The presented XplorerVU is proactive in nature and hence reduces the user interactions for the recommendation generation process by exploiting user's social network profile. The recommendations presented appealingly facilitate the POI selection process and helps to build concrete trip plans. The implicit feedback collection mechanism of XlorerVU reduces the dependencies on changing user preferences, and the user can also provide inputs to update explicit preferences anytime. The overall response of the XplorerVU mobile application is promising with its satisfiable travel recommendations. The travel mode and the financial capabilities of the users are some crucial factors that affect the acceptance ratio of generated recommendations. When the user utilizes the public transport for travel, it can make an impact on the recommended schedule and trip plan. The utilization of intelligent computation models to meet the demands of travelers requires further research to provide excellent travel planning experience to users.

7. Conclusions and future work

The main objective of RSs is to provide interesting and relevant recommendations to the users by knowing their interests. The similarity-based neighborhood determination problem for generating efficient recommendation is a notable problem in recent time RS development. Hence, this article addresses the problem with the proposed QPSO and QICE based collaborative filtering recommender system through an efficient user clustering mechanism. We have attempted to develop a hybrid user clustering model for neighborhood selection problem of RSs and we have almost achieved our goals. The experiments conducted on the two large-scale real-world datasets and the obtained results portray the performance of the proposed QPSO and QICE based CFRS. The proposed recommendation approaches outperform the standalone recommendation approaches with the accurate recommendation in terms of Hit-Rate, Precision, Recall, F-Measure, and Accuracy. From obtained results, an inference is made that the PCC prediction model generates better results compared to the average prediction model on both datasets. The proposed novel clustering ensemble model of QICE with QPSO, k-PSO, and k-MWO has produced better user clustering, and this is an initiative using quantum mechanics and swarm intelligent models for user clustering through an ensemble method. The full utilization of global optimization ability of the algorithms has produced better recommendations. The QICE approach is successfully employed on the proposed XplorerVU mobile recommendation framework and evaluated in the smart cities Tiruchirapalli and Thanjavur through the participant users. The XplorerVU mobile recommendation application is evaluated on the ISO 9241-11-110 standards for the analysis of its user interface and interaction capabilities. The evaluation results have proven its satisfactory performance and positive feedback from the participant users. The proposed QICE based CFRS on XplorerVU is immune to limitations of conventional RSs and specifically, it is not limited to the issues of accuracy and efficiency in the generation of tailored recommendations to the users. As a future work, we intend to incorporate users' online behavior for user clustering, and we will focus on studying the possibilities and impacts on such users clustering models. Further, to enhance the user clustering based recommendation generation process, the nature-inspired intelligent optimization algorithms need to be explored for the combinational optimization problems of recommender systems. We also plan to perform an extensive study to build powerful clustering ensemble model to incorporate user behavior such as users' usage patterns and acceptance of recommendations, for the generation of stable recommendations.

Acknowledgments

The authors are grateful to Science and Engineering Research Board (SERB), Department of Science & Technology, New Delhi, for the financial support (No. YSS/2014/000718/ES). Authors express their gratitude to SASTRA University, Thanjavur, for providing the infrastructural facilities to carry out this research work. We would like to express our gratitude towards the unknown potential reviewers who have agreed to review this paper and provided valuable suggestions to improve the quality of the paper.

References

[1] J. Leino, User Factors in Recommender Systems: Case Studies in E-Commerce, News Recommending, and E-Learning, Tampere, 2014.
[2] B.M. Sarwar, G. Karypis, J. Konstan, J. Riedl, Recommender systems for large scale E-commerce: scalable neighborhood formation using clustering, in: Proceeding Int. Conf. Comput. Inf. Technol. Dhaka, Bangladesh, 2002.
[3] S.-M. Choi, S.-K. Ko, Y.-S. Han, A movie recommendation algorithm based on genre correlations, Expert Syst. Appl. 39 (2012) 8079–8085.
[4] Q. Li, S.H. Myaeng, B.M. Kim, A probabilistic music recommender considering user opinions and audio features, Inf. Process. Manage. 43 (2007) 473–487.
[5] Z. Liu, W. Qu, H. Li, C. Xie, A hybrid collaborative filtering recommendation mechanism for P2P networks, Future Gener. Comput. Syst. 26 (8) (2010) 1409–1417.
[6] S. Cleger-Tamayo, J.M. Fernández-Luna, J.F. Huete, Top-N news recommendations in digital newspapers, Knowl.-Based Syst. 27 (2012) 180–189.
[7] S. Vairavasundaram, V. Varadharajan, I. Vairavasundaram, L. Ravi, Data mining-based tag recommendation system: an overview, Wiley Interdisc. Rew.: Data Min. Knowl. Discov. 5 (2015) 87–112.
[8] A. Abbas, K. Bilal, L. Zhang, S.U. Khan, A cloud based health insurance plan recommendation system: a user centered approach, Future Gener. Comput. Syst. 43–44 (2015) 99–109.
[9] L. Ravi, S. Vairavasundaram, A collaborative location-based travel recommendation system through enhanced rating prediction for the group of users, in: Computational Intelligence and Neuroscience, 2016.
[10] Y. Zhang, M. Chen, D. Huang, D. Wu, Y. Li, iDoctor: Personalized and professionalized medical recommendations based on hybrid matrix factorization, Future Gener. Comput. Syst. 66 (2017) 30–35.
[11] S. Zhang, S. Zhang, N.Y. Yen, G. Zhu, The recommendation system of micro-blog topic based on user clustering, Mob. Netw. Appl. (2017) 1–12.
[12] R. Logesh, V. Subramaniyaswamy, A Reliable Point of Interest Recommendation based on Trust Relevancy between Users, Wireless Personal Communications, 2017.
[13] V. Subramaniyaswamy, R. Logesh, Adaptive KNN Based Recommender System Through Mining of User Preferences, Wireless Personal Communications, 2017.
[14] R. Logesh, V. Subramaniyaswamy, D. Malathi, N. Senthilselvan, A. Sasikumar, P. Saravanan, G. Manikandan, Dynamic particle swarm optimization for personalized recommender system based on electroencephalography feedback, Biomed. Res. (2017).
[15] L. Lü, M. Medo, C.H. Yeung, Y.-C. Zhang, Z.-K. Zhang, T. Zhou, Recommender systems, Phys. Rep. 519 (2012) 1–49.
[16] L. Landau, An Introduction to Recommender Systems, Cambridge University Press, New York, 2011.
[17] B. Sarwar, G. Karypis, J. Konstan, J. Riedl, Item-based collaborative filtering recommendation algorithms, in: Proc. 10th Int. Conf. World Wide, Hong Kong, 2001, pp. 285–295.
[18] D. Li, C. Chen, Q. Lv, L. Shang, Y. Zhao, T. Lu, N. Gu, An algorithm for efficient privacy-preserving item-based collaborative filtering, Future Gener. Comput. Syst. 55 (2016) 311–320.
[19] C.F. Tsai, C. Hung, Cluster ensembles in collaborative filtering recommendation, Appl. Soft Comput. 12 (2012) 1417–1425.
[20] A. Bilge, H. Polat, A comparison of clustering-based privacy-preserving collaborative filtering schemes, Appl. Soft Comput. 13 (2013) 2478–2489.
[21] C. Birtolo, D. Ronca, Advances in clustering collaborative filtering by means of fuzzy C-means and trust, Expert Syst. Appl. 40 (2013) 6997–7009.
[22] D.H. Park, H.K. Kim, I.Y. Choi, J.K. Kim, A literature review and classification of recommender systems research, Expert Syst. Appl. 39 (2012) 10059–10072.
[23] R. Xu, D.C. Wunsch, Survey of clustering algorithms, IEEE Trans. Neural Netw. 16 (3) (2005) 645–678.
[24] L. Wang, M. Rege, M. Dong, Y. Ding, Low-rank kernel matrix factorization for large-scale evolutionary clustering, IEEE Trans. Knowl. Data Eng. 24 (6) (2012) 1036–1050.

[25] M. Bouguessa, S. Wang, Mining projected clusters in high-dimensional spaces, IEEE Trans. Knowl. Data Eng. 21 (4) (2009) 507–522.
[26] R. Jenssen, Mean vector component analysis for visualization and clustering of nonnegative data, IEEE Trans. Neural Netw. Learn. Syst. 24 (10) (2013) 1553–1564.
[27] G. Wilkin, X. Huang, K-means clustering algorithms: implementation and comparison, in: Proceedings IEEE 2nd International Multi-Symposiums on Computer and Computational Sciences, 2007, pp. 133–136.
[28] X. Chen, X. Xu, J. Huang, Y. Ye, TW-k-means: automated two-level variable weighting clustering algorithm for multiview data, IEEE Trans. Knowl. Data Eng. 25 (4) (2013) 932–944.
[29] X.-Z. Gao, J. Wang, J.M.A. Tanskanen, R. Bie, X. Wang, P. Guo, K. Zenger, Optimal classification of epileptic EEG signals using neural networks and harmony search methods, J. Softw. 9 (1) (2014) 230–239.
[30] X.Z. Gao, V. Govindasamy, H. Xu, X. Wang, K. Zenger, Harmony search method: Theory and applications, Comput. Intell. Neurosci. 2015 (2015).
[31] X.-Z. Gao, A.K. Sangaiah, M. Ramachandran, Soft computational approaches for prediction and estimation of software development, Sci. World J. 2016 (2016).
[32] G.-G. Wang, S. Deb, Z. Cui, Monarch butterfly optimization, Neural Comput. Appl. (2015) 1–20.
[33] G.-G. Wang, S. Deb, L.D.S. Coelho, Elephant herding optimization, in: Computational and Business Intelligence (ISCBI), 2015 3rd International Symposium on, IEEE, 2015, pp. 1–5.
[34] G.-G. Wang, S. Deb, L.D.S. Coelho, Earthworm optimization algorithm: a bio-inspired metaheuristic algorithm for global optimization problems, Int. J. Bio-Inspired Comput. (2015c).
[35] G.-G. Wang, S. Deb, A.H. Gandomi, A.H. Alavi, A hybrid PBIL-based Krill Herd Algorithm, in: Computational and Business Intelligence (ISCBI), 2015 3rd International Symposium on, IEEE, 2015d, pp. 39–44.
[36] G.-G. Wang, A.H. Gandomi, A.H. Alavi, S. Deb, A multi-stage krill herd algorithm for global numerical optimization, Int. J. Artif. Intell. Tools 25 (02) (2016) 1550030.
[37] G.-G. Wang, A.H. Gandomi, A.H. Alavi, S. Deb, A hybrid method based on krill herd and quantum-behaved particle swarm optimization, Neural Comput. Appl. 27 (4) (2016) 989–1006.
[38] G.-G. Wang, Moth search algorithm: a bio-inspired metaheuristic algorithm for global optimization problems, Memetic Comput. (2016) 1–14.
[39] B. Balusamy, K. Karthikeyan, A.K. Sangaiah, Ant colony-based load balancing and fault recovery for cloud computing environment, Int. J. Adv. Intell. Paradigms 9 (2–3) (2017) 204–219.
[40] A. Ahmadyfard, H. Modares, Combining PSO and K-means to enhance data clustering, in: Proceedings IEEE International Symposium on Telecommunications, 2008, pp. 688–691.
[41] D. Van der Merwe, A.P. Engelbrecht, Data clustering using particle swarm optimization, in: Proceedings IEEE Congress on Evolutionary Computation, vol. 1, 2003, pp. 215–220.
[42] S. Rana, S. Jasola, R. Kumar, A boundary restricted adaptive particle swarm optimization for data clustering, Int. J. Mach. Learn. Cybern. 4 (2013) 391–400.
[43] X. Chen, A new clustering algorithm based on near neighbor influnece, Expert Syst. Appl. 42 (2015) 7746–7758.
[44] S. Huang, H. Wang, D. Li, Y. Yang, T. Li, Spectral co-clustering ensemble, Knowl.-Based Syst. 84 (2015) 46–55.
[45] L. Jing, K. Tian, J.Z. Huang, Stratified feature sampling method for ensemble clustering of high dimensional data, Pattern Recognit. 48 (2015) 3688–3702.
[46] Z. Zhang, D. Pati, A. Srivastava, Bayesian clustering of shapes of curves, J. Stat. Plann. Infer. 166 (2015) 171–186.
[47] C. Qin, S. Song, C. Huang, L. Zhu, Unsupervised neighborhood component analysis for clustering, Neurocomputing 168 (2015) 609–617.
[48] L.N. Ferreira, L. Zhao, Time series clustering via community detection in networks, Inf. Sci. 326 (2016) 227–242.
[49] W. Cai, A manifold learning framework for both clustering and classification, Knowl. Based Syst. 89 (2015) 641–653.
[50] D. Huang, J.H. Lai, C.D. Wang, Combining multiple clustering via crowd agreement estimation and multi-granularity link analysis, Neurocomputing 170 (2015) 240–250.
[51] I. Saha, J.P. Sarkar, U. Maulik, Ensemble based rough fuzzy clustering for categorical data, Knowl. Based Syst. 77 (2015) 114–127.
[52] Z. Liu, Q. Pan, J. Dezert, G. Mercier, Credal c-means clustering method based on belief functions, Knowl. Based Syst. 74 (2015) 119–132.
[53] A.E. Gutierrez-Rodriguez, J. Fco Martinez-Trinidad, M. Garcia-Borroto, J.A. Carrasco-Ochoa, Mining patterns for clustering on numerical datasets using unsupervised decision trees, Knowl. Based Syst. 82 (2015) 70–79.
[54] J. Sun, W. Chen, W. Fang, X. Wun, W. Xu, Gene expression data analysis with the clustering method based on an improved quantum-behaved Particle Swarm Optimization, Eng. Appl. Artif. Intell. 25 (2) (2012) 376–391.
[55] W.Y. Hsu, Application of quantum-behaved particle swarm optimization to motor imagery EEG classification, Int. J. Neural Syst. 23 (06) (2013) 1350026.
[56] D. Goldberg, D. Nichols, B.M. Oki, D. Terry, Using collaborative filtering to weave an information tapestry, Commun. ACM 35 (1992) 61–70.
[57] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, J. Riedl, Grouplens: An open architecture for collaborative filtering of netnews, in: Proc. Conf. Comput. Supported Cooperative Work, 1994, pp. 175–186.
[58] G. Adomavicius, A. Tuzhilin, Toward the next generation of recommendation system: a survey of the state-of-the-art and possible extensions, IEEE Trans. Knowl. Data Eng. 17 (6) (2005) 734–749.
[59] X. Amatriain, J. Basilico, Netflix Recommendations: Beyond the 5 Stars (Part 1), Netflix Tech Blog, 2012.
[60] M. Deshpande, G. Karypis, Item-based top-N recommendation algorithms, ACM Trans. Inf. Syst. 22 (1) (2004) 143–177.
[61] Y. Koren, R. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, IEEE Comput. 42 (8) (2009) 30–37.
[62] C. Basu, H. Hirsh, W. Cohen, Recommendation as classification: Using social and content-based information in recommendation, in: Proc. Conf. Artif. Intell. 1998, pp. 714–720.
[63] R. Burke, Hybrid recommender systems: Survey and experiments, User Model. User-Adapt. Interact. 12 (4) (2002) 331–370.
[64] Y. Koren, The bellkor solution to the netflix grand prize (2009) [Online]. Available: http://www.netflixprize.com/assets/GrandPrize2009_BPC_BellKor.pdf.
[65] I. Pilaszy, D. Tikk, Recommending new movies: Even a few ratings are more valuable than metadata, in: Proc. ACM Conf. Recommender Syst. 2009, pp. 93–100.
[66] P.H. Abreu, D.C. Silva, F. Almeida, J. Mendes-Moreira, Improving a simulated soccer team's performance through a memory-based collaborative filtering approach, Appl. Soft Comput. J. 23 (2014) 180–193.
[67] P.H. Abreu, D.C. Silva, J. Portela, J. Mendes-Moreira, L.P. Reis, Using model-based collaborative filtering techniques to recommend the expected best strategy to defeat a simulated soccer opponent, Intell. Data Anal. 18 (5) (2014) 973–991.
[68] K. Ericson, S. Pallickara, On the performance of high dimensional data clustering and classification algorithms, Future Gener. Comput. Syst. 29 (2012) 1024–1034.
[69] Z. Wang, X. Yu, N. Feng, Z. Wang, An improved collaborative movie recommendation system using computational intelligence, Vis. Lang. Comput. 25 (2014) 667–675.
[70] A. Gupta, B.K. Tripathy, A generic hybrid recommender system based on neural networks, in: IEEE Int. Adv. Comput. Conf., IEEE, 2014, pp. 1248–1252.
[71] L. Cheng, H. Wang, A fuzzy recommender system based on the integration of subjective preferences and objective information, Appl. Soft Comput. 18 (2014) 290–301.
[72] C. Birtolo, D. Ronca, R. Armenise, Improving accuracy of recommendation system by means of item-based fuzzy clustering collaborative filtering, in: Int. Conf. Intell. Syst. Des. Appl. ISDA, 2011, pp. 100–106.
[73] L. Gong, T. Xu, W. Zhang, X. Li, X. Wang, W. Pan, The application of mixed recommendation algorithm with user clustering in the microblog advertisements promotion, in: AIP Conference Proceedings, vol. 1820, no. 1, AIP Publishing, 2017, p. 080025.
[74] C. Bouras, V. Tsogkas, Improving news articles recommendations via user clustering, Int. J. Mach. Learn. Cybern. (2017) 1–15.
[75] X. Wang, J. Liu, The research of broadcast television program recommendation technology based on user clustering, in: Computational Intelligence and Design (ISCID), 2016 9th International Symposium on, vol. 2, IEEE, 2017, pp. 403–407.
[76] X. Zheng, D. An, X. Chen, W. Guo, Interest prediction in social networks based on markov chain modeling on clustered users, Concurr. Comput. Pract. Exp. (2017).
[77] H. Zhang, X. Zhang, X.-Z. Gao, S. Song, Self-organizing multiobjective optimization based on decomposition with neighborhood ensemble, Neurocomputing 173 (2016) 1868–1884.
[78] T.M. Silva Filho, B.A. Pimentel, R.M. Souza, A.L. Oliveira, Hybrid methods for fuzzy clustering based on fuzzy c-means and improved particle swarm optimization, Expert Syst. Appl. 42 (17) (2015) 6315–6328.
[79] H. Koohi, K. Kiani, User Based Collaborative Filtering using Fuzzy C-Means. Measurement, 2016.
[80] N. Pal, K. Pal, J. Keller, J. Bezdek, A possibilistic fuzzy c-means clustering algorithm, IEEE Trans. Fuzzy Syst. 13 (4) (2005) 517–530.
[81] J.C. Bezdek, R. Ehrlich, W. Full, FCM: The fuzzy c-means clustering algorithm, Comput. Geosci. 10 (2) (1984) 191–203.

[82] L. Zhang, W. Pedrycz, W. Lu, X. Liu, L. Zhang, An interval weighed fuzzy c-means clustering by genetically guided alternating optimization, Expert Syst. Appl. 41 (13) (2014) 5960–5971.
[83] H. Izakian, A. Abraham, Fuzzy c-means and fuzzy swarm for fuzzy clustering problem, Expert Syst. Appl. 38 (3) (2011) 1835–1838.
[84] N. Nedjah, R.M. Calazan, L.M. Mourelle, C. Wang, Parallel implementations of the cooperative particle swarm optimization on many-core and multi-core architectures, Int. J. Parallel Program. 44 (6) (2016) 1173–1199.
[85] S. Alam, G. Dobbie, Y.S. Koh, P. Riddle, S.U. Rehman, Research on particle swarm optimization based clustering: A systematic review of literature and techniques, Swarm Evol. Comput. 17 (2014) 1–13.
[86] C. Li, J. Zhou, P. Kou, J. Xiao, A novel chaotic particle swarm optimization based fuzzy clustering algorithm, Neurocomputing 83 (2012) 98–109.
[87] S. Chen, Z. Xu, Y. Tang, A hybrid clustering algorithm based on fuzzy cmeans and improved particle swarm optimization, Arab. J. Sci. Eng. 39 (12) (2014) 8875–8887.
[88] Y. Zhang, X. Xiong, Q. Zhang, An improved self-adaptive PSO algorithm with detection function for multimodal function optimization problems, Math. Probl. Eng. (2013).
[89] Z. Pei, X. Hua, J. Han, The clustering algorithm based on particle optimization algorithm, in: Proceedings International Conference on Intelligent Computation Technology and Automation, vol. 1, 2008, pp. 148–151.
[90] J. An, Q. Kang, L. Wang, Q.D. Wu, Mussels wandering optimization: an ecologically inspired algorithm for global optimization, Cogn. Comput. 5 (2) (2013) 188–199.
[91] W. Fang, J. Sun, Z.-P. Xie, W.-B. Xu, Convergence analysis of quantum-behaved particle swarm optimization algorithm and study on its control parameter, Wuli Xuebao/Acta Phys. Sin. 59 (6) (2010) 3686–3694.
[92] W. Song, Y. Qiao, S.C. Park, X. Qian, A hybrid evolutionary computation approach with its application for optimizing text document clustering, Expert Syst. Appl. 42 (5) (2015) 2517–2524.
[93] L. Song, K.K. Chai, Y. Chen, J. Loo, S. Jimaa, J. Schormans, QPSO-based energy-aware clustering scheme in the capillary networks for Internet of Things systems, in: Wireless Communications and Networking Conference, WCNC, 2016 IEEE 2016, pp. 1–6.
[94] C. Domeniconi, M. Al-Razgan, Weighted cluster ensembles: methods and analysis, ACM Trans. Knowl. Discov. Data. 2 (4) (2009) 1–40.
[95] X.P. Fan, Y.S. Xie, Z.F. Liao, A Weighted cluster ensemble algorithm based on graph, in: Proceedings IEEE 10th International Conference on Trust, Security and Privacy in Computing & Communications, 2011, pp. 1519–1523.
[96] N. Iam-On, T. Boongeon, S. Garrett, C. Price, A link-based cluster ensemble approach for categorical data clustering, IEEE Trans. Knowl. Data Eng. 24 (3) (2012) 413–425.
[97] Z.H. Zhou, Ensemble Methods: Foundations and Algorithms, CRC Press, Boca Raton, FL, 2012.
[98] A. Strehl, J. Ghosh, Clustering ensembles-a knowledge reuse framework for combining multiple partitions, J. Mach. Learn. Res. 3 (2002) 583–617.
[99] J. Jia, X. Xiao, B. Liu, Similarity-based spectral clustering ensemble selection, in: Proceedings IEEE 9th International Conference on Fuzzy Systems and Knowledge Discovery, 2012, pp. 1071–1074.
[100] G. Zhu, J. Cao, C. Li, Z. Wu, A recommendation engine for travel products based on topic sequential patterns, Multimedia Tools Appl. (2017) 1–18.
[101] B. Valeri, F. Casati, F. Daniel, On the value of purpose-orientation and focus on locals in recommending leisure activities, J. Web Eng. 16 (1–2) (2017) 63–74.
[102] I. Cenamor, T. de la Rosa, S. Núñez, D. Borrajo, Planning for tourism routes using social networks, Expert Syst. Appl. 69 (2017) 1–9.
[103] T. De Pessemier, J. Dhondt, L. Martens, Hybrid group recommendations for a travel service, Multimedia Tools Appl. 76 (2) (2017) 2787–2811.
[104] F. Santos, A. Almeida, C. Martins, P. Oliveira, R. Gonçalves, Tourism recommendation system based in user functionality and points-of-interest accessibility levels, in: International Conference on Software Process Improvement, Springer International Publishing, 2016, pp. 275–284.
[105] D. Gavalas, C. Konstantopoulos, K. Mastakas, G. Pantziou, A survey on algorithmic approaches for solving tourist trip design problems, J. Heuristics 20 (3) (2014) 291–328.
[106] D. Gavalas, V. Kasapakis, C. Konstantopoulos, G. Pantziou, N. Vathis, C. Zaroliagis, The ecompass multimodal tourist tour planner, Expert Syst. Appl. 42 (21) (2015) 7303–7316.
[107] Y. Han, H. Guan, J. Duan, Tour route multiobjective optimization design based on the tourist satisfaction, Discrete Dyn. Nat. Soc. 2014 (2014).
[108] I.R. Brilhante, J.A. Macedo, F.M. Nardini, R. Perego, C. Renso, On planning sightseeing tours with tripbuilder, Inform. Process. Lett. 51 (2) (2015) 1–15.
[109] Y. Kurata, T. Hara, CT-Planner4: Toward a more user-friendly interactive day-tour planner, in: Proceedings of the 2014 International Conference on Information and Communication Technologies in Tourism, 2014, pp. 73–86.
[110] T. Hasuike, H. Katagiri, H. Tsubaki, H. Tsuda, Route planning problem with groups of sightseeing sites classified by tourist's sensitivity under time expanded network, in: Proceedings of the 2014 IEEE International Conference on Systems, Man and Cybernetics, 2014, pp. 188–193.
[111] J. Liang, L. Bai, C. Dang, The k-means-type algorithms versus imbalanced data distributions, IEEE Trans. Fuzzy Syst. 20 (4) (2012) 728–745.
[112] Q. Kang, S. Liu, M. Zhou, S. Li, A weight-incorporated similarity-based clustering ensemble method based on swarm intelligence, Knowl.-Based Syst. 104 (2016) 156–164.
[113] D. Almazro, G. Shahatah, L. Albdulkarim, M. Kherees, R. Martinez, W. Nzoukou, A Survey Paper on Recommender Systems, 2010. arXiv:1006.5278.
[114] J. Bobadilla, F. Ortega, A. Hernando, A. Gutiérrez, Recommender systems survey, Knowl.-Based Syst. 46 (2013) 109–132.
[115] M.Y.H. Al-shamri, Power coefficient as a similarity measure for memory-based collaborative recommender systems, Expert Syst. Appl. 41 (2014) 5680–5688.
[116] A. Zenebe, A.F. Norcio, Representation, similarity measures and aggregation methods using fuzzy sets for content-based recommender systems, Fuzzy Sets and Systems 160 (2009) 76–94.
[117] J. Bobadilla, F. Ortega, A. Hernando, A collaborative filtering similarity measure based on singularities, Inf. Process. Manage. 48 (2012) 204–217.
[118] H. Wen, G. Ding, C. Liu, J. Wang, Matrix factorization meets cosine similarity: addressing sparsity problem in collaborative filtering recommender system, Web Technol. Appl. (2014) 306–317.
[119] C. Rana, S. Kumar, An extended evolutionary clustering algorithm for an adaptive recommender system, Soc. Netw. Anal. Min. 4 (2014) 1–13.
[120] M. Ramezani, P. Moradi, F. Akhlaghian, A pattern mining approach to enhance the accuracy of collaborative filtering in sparse data domains, Physica A 408 (2014) 72–84.
[121] L. Duan, H. Tian, A practical approach for financial services discovery based on user clustering, ICIC Express Lett. B Appl. Int. J. Res. Surv. 7 (11) (2017) 2419–2424.

Logesh R. received the B.Tech. degree in Computer Science & Engineering and M.Tech. degree in Networking from Pondicherry University, India. Currently, he is a Junior Research Fellow and pursuing the Ph.D. degree in the School of Computing, SASTRA University, India. His research interests include Recommender Systems, Information Retrieval, and Human–Computer Interaction.

Subramaniyaswamy V. received the B.E. degree in Computer Science & Engineering and M.Tech. degree in Information Technology from Bharathidasan University, India and Sathyabama University, India respectively. He received his Ph.D. degree from Anna University, India and currently working as a Senior Assistant Professor in the School of Computing, SASTRA University, India. He has published more than 50 papers in reputed international journals and conferences. His research interests include Data Mining, Recommender Systems, and Big Data Analytics.

Vijayakumar V. is currently working as Professor in VIT University Chennai Campus; He has more than 18 years of experience which includes 8 years in teaching and 8 years in the Industry. He is also a coordinator of Cloud Computing Research Group and Coordinator for Internship in India and Worldwide. In VIT, he involved in many Research and Development Activities, he has also organized many National/International Seminars/ Workshops/ Symposiums/ Conferences/Special Sessions in the area of Cloud computing and Big Data which includes ISBCC'14, ISBCC'15, ISBCC'16 in India, CCCA'14 in Vietnam and CCNC'14 in USA. His area of research includes Grid Computing, Cloud computing, Big Data, Web semantics and also involved in the domain like biomedical application, Mammogram, Autism, Immune system and other areas like key management, security issues in cloud and grid computing.

Xiao-Zhi Gao earned his D.Sc. (Tech.) degree from the Helsinki University of Technology (now Aalto University), Finland in 1999. He was appointed as a Docent (Adjunct Professor) at the same university in 2004. Since October 2016, he has been working at the Lappeenranta University of Technology, Finland. He is also a Guest Professor/Visiting Professors at several universities in China. He has published more than 300 technical papers on refereed journals and international conferences, and his current Google Scholar h-index is 21. His research interests are nature-inspired computing methods (e.g., neural networks, fuzzy logic, evolutionary computing, swarm intelligence, and artificial immune systems) with their applications in optimization, data mining, control, fault diagnosis, signal processing, and industrial electronics.

Indra Gandhi V. completed B.E. in Electrical and Electronics Engineering from Bharathidasan University in the year 2004. She received M.E. in Power Electronics and Drives from Anna University and awarded Gold Medal for the achievement of University first rank. Subsequently, she is awarded with doctorate of philosophy at Anna University, Chennai in the year 2015. At present she is serving as an Associate Professor in the School of Electrical Engineering, VIT University, Vellore, Tamilnadu. She has been engaged in research work for the past 8 years in the area of Power Electronics, Renewable Energy Systems and Computational Intelligence.