Mobile Networks and Applications (2019) 24:1226–1239
https://doi.org/10.1007/s11036-019-01260-4

Hybrid Location-based Recommender System for Mobility and Travel Planning

Logesh Ravi [1] · V. Subramaniyaswamy [1] · V. Vijayakumar [2] · Siguang Chen [3] · A. Karmel [2] · Malathi Devarajan [1]

Published online: 19 April 2019

Abstract

In recent times, the modern developments of internet technologies and social networks have attracted global researchers to explore the recommender systems for generating personalized location-based services. Recommender Systems (RSs) as proven decision support tools have gained immense popularity to solve information overloading problem among various real-time applications of e-commerce, travel and tourism, movies and e-learning. RSs emerge as a popular and reliable information filtering approach that is capable of suggesting relevant items, movies, and locations to the active target user based on dynamic preferences and interests. Beyond the development of many feature-rich recommendation algorithms, the need for a better full-fledged RS to produce precise and highly relevant recommendations based on ratings and preferences provided by the target user is very high. With the specific focus to the travel domain, the global research community has been involved in the development of a complete travel recommender system that is immune to the sparsity and cold start problems. In this paper, we present a new Hybrid Location-based Travel Recommender System (HLTRS) through exploiting ensemble based co-training method with swarm intelligence algorithms to enhance the personalized travel recommendations. The proposed HLTRS is experimentally validated on the real-world large-scale dataset, and we have made an extensive user study to determine the ability of developed RS to produce user satisfiable recommendations in real-time scenarios. The obtained results and analyses demonstrate the improved performance of the proposed Hybrid Location-based Travel Recommender System over existing baselines of recommender systems research.

Keywords Recommender systems · Point-of-interest · Travel recommendation · Personalization · Travel planning

1 Introduction

Recommender System (RS) has emerged as an effective decision support mechanism to tackle information overloading problem faced by digital users during the search of relevant interesting items that meet their changing requirements in recent years [1]. In various real-time applications such as movies, music, books, tourism, e-commerce, and education, Recommender Systems are widely used [2–7]. Among various techniques developed for making personalized recommendations, Collaborative Filtering, Content-based Filtering, and Hybrid Filtering were widely utilized to support active target users by providing a personalized list of recommendations. The rapid increase in the utilization of Smartphones and social networks has resulted with the huge amount of user-generated data, and this data is considered to be a significant

✉ V. Subramaniyaswamy
vsubramaniyaswamy@gmail.com

Logesh Ravi
LogeshPhD@gmail.com

V. Vijayakumar
vijayakumar.v@vit.ac.in

Siguang Chen
sgchen@njupt.edu.cn

A. Karmel
karmel.a@vit.ac.in

Malathi Devarajan
malathipuducherry@gmail.com

1 School of Computing, SASTRA Deemed University, Thanjavur, India

2 School of Computing Science and Engineering, Vellore Institute of Technology, Chennai, India

3 Key Lab of Broadband Wireless Communication and Sensor Network Technology of Ministry of Education, Nanjing University of Posts and Telecommunications, Nanjing, China

source for the recommender systems to produce personalized recommendation list [8]. Figure 1 depicts the overview of the recommendation generation process in the recommender systems. Online users share their thoughts through social network and recommendations are presented by considering target users' social behaviors and activities.

The growing development of web-based technologies and social networks, users tend to share their opinions and thoughts online [9]. As location-based social network helps to share their experiences, it is essential to built users' profile based on social activities and behavior to learn more about users' preferences from historical check-ins [10, 11]. Travel recommender system suggests point of interests to the user by mining users' interests and preferences implicitly or explicitly from location-based social networks. Based on users' historical check-ins, similar users are identified and rating is predicted for the unknown places. To provide personalized recommendations in real-time applications, a good recommender system should consider user requirements, social ties, geographical features and contextual information to predict highly relevant items which are more probably to be accepted by the target user [12]. Though collaborative filtering based recommender system alleviates information overloading problem, they are still suffering from cold start, sparsity and personalization problems. To address the aforementioned issues, most researchers have used opinions from trusted users to generate personalized recommendations rather than from similar users [13].

In collaborative filtering based recommender system, different clustering algorithms were used to discover similar users. But, the traditional clustering algorithm has some drawbacks to yield optimal solutions [14]. To conquer such limitations, swarm intelligent algorithms have been introduced for effective user clustering. The emergence of swarm intelligence algorithms has attracted many researchers to exploit such algorithms in recommender system for user clustering [15]. The nature-inspired meta-heuristic algorithms such as cuckoo search, particle swarm optimization and ant colony optimization algorithms were utilized in various applications to tackle complex optimization problems [16–18]. Bio-inspired algorithm inherits the features of the biological system and yields promising convergence results [19]. Especially in travel recommendations, both the number of travellers and point of interests are large, but user rating is very low which leads to data sparsity problem. Hence correlation between travellers cannot be derived appropriately, and so it may not be reliable. Similarly, the problem of cold start arises when travel recommender system tries to generate recommendations for the unfamiliar point of interests or for new travellers [20]. Thus the recommender system gains a negative impression on the performance of traditional filtering algorithms. It will be more critical to generate personalized recommendations as it contains only very few rating information.

Recently, Da Costa et al. [21] proposed Co-training based recommender system (CoRec) to resolve cold start and sparsity problem by strengthening the rating matrix as a pre-processing step. In 1998, Blum and Mitchell introduced the concept of Co-training as a distinct multi-view learning model [22]. Co-training algorithm uses multiple views of a single feature set that are uncorrelated by its description and compatible if all are identically labeled. As an extension ECo-Rec is proposed, where CoRec is generalized to work with more than two recommendation algorithms and aggregate the results of simultaneously co-trained models to generate consensus user-item rating matrix [23]. Emerging demand for personalized travel recommendations broadens the scope for the significant progress of location-based travel recommender system. We have utilized the co-training method in the travel recommendation process to handle user-POI sparsity and cold start problem of traditional recommender system by predicting ratings for unfamiliar POIs and new user. The proposed HLTRS is validated on the TripAdvisor dataset to demonstrate the applicability of the proposed travel recommendation generation process in

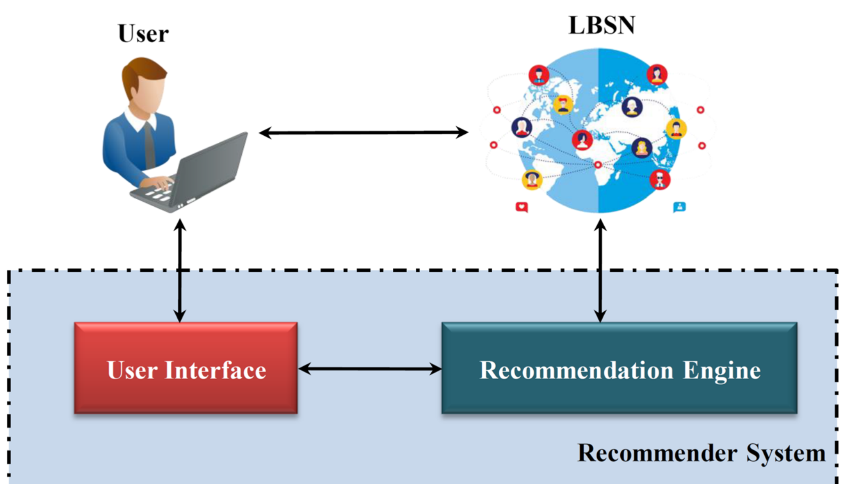


Fig. 1 Overview of the recommender system and recommendation generation process

real time scenarios. The major contributions of the proposed HLTRS are summarized as follows.

- We have developed a personalized travel recommender system by analyzing target users' online social activities and behaviors.
- We have employed ensemble based co-trained swarm intelligence algorithms to identify the best neighborhood for rating prediction and to enhance the rating matrix.
- Contextual information has been utilized to minimize the computational complexity and dimensionality reduction of the recommendation generation process.
- The proposed HLTRS is experimentally validated on the real time complex dataset to exhibit the performance and efficiency over existing baselines.

The remainder of the article is structured as follows: In section 2, recent literature related to travel recommender system, collaborative filtering algorithms, social user profiling, swarm intelligence for effective user clustering and co-training method for rating enhancement are presented. In section 3, the detailed explanation of the proposed Hybrid Location-based Travel Recommender System (HLTRS) is presented. Section 4 portrays the results obtained from experiments and discusses the performance efficiency of the proposed HLTRS over other existing recommendation models. Finally, in section 5, we conclude the work done along with future research directions.

2 Related works

Recently, the recommender system has been widely explored in various application domains with different methodologies to strengthen the performance efficiency of the recommendation generation process. The overview of the conventional recommender system framed using collaborative filtering techniques; advantages of using swarm intelligent algorithms in the recommender system, trust-aware recommendations, and co-training method for rating enhancement are presented in the following subsections.

2.1 Recommender systems based on collaborative filtering

The most generally used filtering algorithm in the recommender system is collaborative filtering. Collaborative filtering algorithm predicts the rating for unknown items of the target user based on the ratings of the items provided by the neighbors [24]. Goldberg et al. [25] first established the idea of collaborative filtering to discover similar users and now it is widely used technique in the recommender systems. Later, in 1997, GroupLens [26] utilized the collaborative filtering algorithm to group the news articles automatically in Usenet. Herlocker et al. [27] present the measuring factors to estimate the quality of the rating prediction mechanism of recommender systems. Despite the benefits provided by the collaborative filtering algorithm, cold start and data sparsity are the major concern of the conventional recommender systems. Though some existing collaborative filtering based recommender systems provide better results, at most times the contextual information and user preferences are not considered to produce recommendations [28]. These collaborative filtering algorithms sometimes necessitate domain experts for valuable suggestions.

2.2 Trust-aware recommendations

As the collaborative filtering based recommender system is vulnerable to malicious attacks [29, 30] and data sparsity problem, incorporating trust information helps to conquer such issues. This is due to the fact that people used to buy items suggested by friends, family members and colleagues whom they believe the most. Lathia et al. [31] have proposed the collaborative filtering based recommender system with the trusted k-nearest neighbor to alleviate the problems of the traditional recommender system. Massa and Avesani [32] used four different trust metrics to alleviate ratings sparsity and cold start problems of the conventional recommender systems. Kant and Bharadwaj [33] exploit both trust and untrust information in the recommender system to enhance the recommendation quality through the fuzzy model. Guo et al. [34] performed experimental research on trust metrics of the recommender system and stated that each trust metric has its advantages and limitations depend on the context of usage. Gupta and Nagpal [35] present the survey on trust metrics and its properties for their application in various contexts. As recommendation generation process grows, trust metrics need to be recomputed to produce good results.

2.3 Swarm intelligent algorithms in recommender systems

Various clustering algorithms were used in collaborative filtering based recommender system to discover neighbor sets for rating predictions. Due to the drawback of the conventional clustering algorithm to obtain optimal solutions on large scale applications, many researchers have incorporated swarm intelligence algorithms in various recommender systems for user clustering. Swarm intelligent algorithms inherit the features of biological systems to provide better convergence rate. Ujjin and Bentley [36] exploit particle swarm intelligent algorithm to produce tailored recommendations using a modified Euclidean distance. The experiments were conducted on MovieLens datasets, and showed that PSO performs better than the genetic algorithm. Katarya and Verma [37] developed a recommendation model by combining particle swarm optimization and fuzzy c-means algorithm to generate personalized movie recommendations with reduced computational cost. To find weights for features of Jester dataset, Choudhary et al. [38] utilized Gravitational search algorithm (GSA) and showed that GSA outperforms PSO. Bedi and Sharma [39] employed Ant colony optimization (ACO) for trust computation, and it solves the problem like local optima. Likewise, meta-heuristic algorithms such as Mussels wandering optimization (MWO) [40] and Cuckoo search algorithm (CSA) [41] are used to address the limitations of traditional clustering such as k-means. The integration of k-means with MWO and CSO provides better clustering results than individual algorithms. In general, every clustering algorithm has its limitations and thus integrating results of two or more hybrid clustering algorithms helps to generate promising solutions [42, 43].

2.4 Travel recommender system

Travel Recommender system has gained significant interest from many researchers due to the employment of swarm intelligence for user clustering algorithms and personalized recommendation generations [44–46]. The major goal of the travel recommender system is to satisfy users' requirements. Various parameters such as user preference, interest, weather, time of the day, companion and travel costs are considered as some constraints while making recommendations. Based on the target users' requirements, the influential features of the travel recommender system are adjusted to generate the relevant point of interests. Logesh et al. [47] proposed a recommendation model called DPSOHiK for personalized point of interest recommendations based on hybrid particle swarm optimization algorithm using electroencephalography feedback. Brilhante et al. [48] developed a web-based travel recommendation model called TripBuilder. They acquired the information from Wikipedia and Flickr dataset to evaluate the performance of TripBuilder. Similarly, Kurata and Hara [49] proposed a travel plan recommender called CT-Planner by exploiting genetic algorithms. Notably, most of the available travel recommender system facilitates general recommendations and lack personalization. Logesh and Subramaniyaswamy [50] developed a hybrid travel recommendation model called PCAHTRS using contextual data to facilitate personalized recommendations.

2.5 Co-training method

The challenging issue of any recommender system is to deal with large user-item matrix where most of the entries are empty [51]. One of the most useful ways to handle rating sparsity and cold start difficulty of the conventional recommender system is to fill those missing values. Recently, some researchers have utilized the co-training method in the recommender system to enrich the rating matrix. Zhang et al. [52] presented a context-aware semi-supervised co-training model called CSEL to tackle the cold start problem of the recommender system. Zhang and Wang [53] developed a unified Collaborative Multi-view Learning (CML) model to alleviate the sparsity problem of recommender system by providing an additional view to strengthen the sparse user-item rating matrix. Quang et al. [54] utilized a co-training based collaborating filtering method that iteratively extends the training set by exchanging user and item features to provide enhanced prediction accuracy. More recently, Matuszyk and Spiliopoulou [55] introduced a semi-supervised stream based recommendation model with the co-training approach. Here the prediction model generates reliable predictions and uses the same as a label to improve the recommendation generation process. To yield enhanced recommendations, the results obtained from individual algorithms are joined together as an ensemble to generate consensus rating matrix for rating prediction [56].

3 Proposed hybrid location-based travel recommender system

The most important aim of the proposed HLTRS is to make the search process of the tourist simpler and more convenient. The proposed HLTRS generates POIs and travel routes which are more specific to the target user. Based on the historical check-ins, interests, preferences, and requirements, personalized recommendations are generated. Though lots of travel recommender systems are available, they suffer from cold start and data sparsity problem. Despite, existing travel recommender system facilitates only general recommendations and not user-specific or personalized recommendations. Travel Recommender systems are decision-making tools and different from other recommender systems, analyzing users' interests and preferences are considered as an essential factor to construct efficient user-centric recommender system. The major goal of the proposed HLTRS is to generate personalized recommendations for travellers by exploiting social network data. Information such as historical check-ins, social ties, feedbacks and reviews about already visited sites, social activities and behavior are acquired from users' social network. By mining such information and building user profiles, it becomes easier to develop an efficient personalized travel recommendation model.

The proposed HLTRS is designed for travellers to facilitate personalized recommendations based on users' social activities and preferences. Our proposed Hybrid Location-based Travel Recommender System builds user profile from social network data, and contextual data are utilized to remove irrelevant information. To strengthen the user-POI rating matrix, an ensemble based co-trained swarm intelligence method is employed for user clustering, and finally personalized recommendations are presented to the target user. The recommendation generation process of the proposed HLTRS is portrayed in Fig. 2. Initially, the user profile is constructed to learn about target users' social activities and behavior to boost the recommendation generation process. Based on the profile built from social data, the user is clustered into a group from where nearest neighbors are selected for rating prediction. In HLTRS, Co-training technique is used to strengthen the user-POI rating matrix for accurate rating prediction. It significantly reduces the rating sparsity and cold start difficulty of the conventional travel recommender system. Then the rating predicted POIs are ranked and top ranked POIs are suggested as the personalized recommendations.

3.1 User profiling

The proposed HLTRS aims to provide personalized travel recommendations on the basis of users' interests, preferences, and requirements. To learn more about the preferences of the target user, the user profile is constructed from the social activities and behaviors collected from location based social network data. User profiles are modeled with respect to demographic features and preferences. Demographic features consist of users' personal details such as age, gender, and employment, whereas preferences consist of users' spatial-temporal features such as historical check-ins, reviews, and feedbacks, comments, posts, and requirements. A users' preference on POI varies in association with weather, season of the year, time preferred to travel and with whom they wish to travel like family or friends, interests and travel cost. For instance, people usually go for sightseeing in the morning and evening, and have food at noon. Thus, we utilized location-based social network data to extract the influential features of the user check-ins.

The multi-layer user profiling model based on online reviews and ratings are employed in HLTRS. Each layer symbolizes the user preference with respect to influential feature sets such as weather, cost estimation, companion, travel frequency and time. Only positive reviews of the user feedback are used to build a user profile. The reason behind choosing only positive reviews is to discover and suggest the point of interests that are more relevant to the users' interests. If the rating provided by the target user is lesser than 3 stars, it is considered as a negative review and ignored, while others are considered as a positive review. Since traveler used to rate their experience in the form of "five stars", they are used to predict the rating for unfamiliar point of interests. The multilayer user profile θ_u built by language model [57] is defined as follows

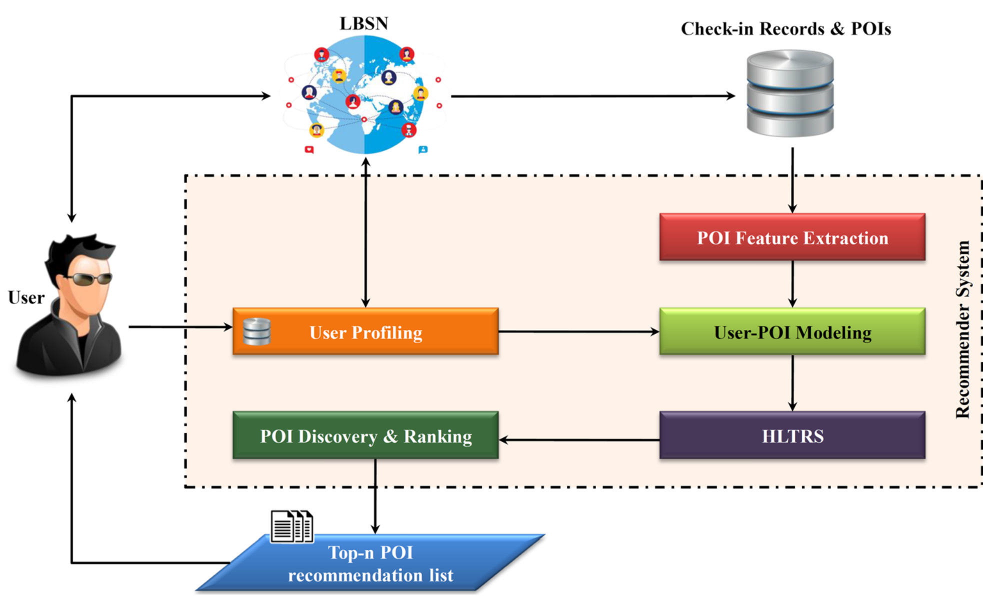
$$\theta_u = \frac{\sum_{r_i \in R_f} p(w|r_i)}{|R_f|} \tag{1}$$

where $R_f = \cup r_i$ is the union of all positive reviews about the feature i, w represents the influential word in the review and $p(w|r_i)$ represents the conditional probability estimated as follows

$$p(w|r_i) = \frac{F_w^{r_i} + \mu \frac{F_w^{R_f}}{\sum_w F_w^{R_f}}}{\sum_w F_w^{r_i} + \mu} \tag{2}$$

where μ is the smoothing parameter, $F_w^{r_i}$ is the influential word w frequency of r_i and $F_w^{R_f}$ is the influential word w frequency of R_f.

Fig. 2 Recommendation generation process of the proposed HLTRS



3.2 Trust utilization for better personalization

The users' rating profile has various features to predict the point of interests for the target user. But users' trust network does not provide such features to discover similar neighbor sets of the target user. Thus, measuring the performance efficiency of both the rating profile and trust networks helps to predict the point of interests more accurately. In Hybrid Location-based Travel Recommender System, user reliability is computed to identify the user with low rating profile and trust network with an intention to strengthen the rating profile. Larger the number of available ratings, the probability of predicting the relevant point of interests for that user is also high. This is due to the fact that a large number of similar users is identified and by estimating the similarity score between users, there exists a high probability to generate more relevant POIs. Finally, ratings are predicted more accurately for unfamiliar point of interests and recommendation generated are more probably accepted by the target user. Similarly, the cardinality of a neighbor set and the cardinality of ratings provided by that neighbors also influence the estimation of user reliability. The summation of the similarities computed between the target user and similar users helps to strengthen the performance of the rating prediction algorithms.

3.3 Swarm intelligence for user clustering

With the growing popularity of online users and the web-based services, the need for personalized travel recommender system increases. We incorporate swarm intelligence algorithms for user clustering and ratings are predicted for the POIs from the consensus similarity matrix obtained from the co-trained swarm intelligent algorithms for better travel recommendations. The proposed HLTRS exploits three swarm intelligent algorithms such as Mussels wandering, ant colony and Cuckoo search optimization algorithms and for user clustering. The HLTRS is designed to cluster the input dataset independently by swarm intelligent algorithms, and then the resulted similarity matrix is used as additional information for other swarm intelligent algorithms for user clustering. Thus a swarm intelligent algorithm not only learns from its model but also from the other algorithms for efficient clustering. The process continues for some iteration to generate better user-POI rating matrix. Larger the number of iterations means higher the prediction accuracy. Finally, the similarity matrices obtained from co-trained swarm intelligent algorithms are combined to generate a consensus similarity matrix through an ensemble approach [58]. From the consensus matrix, ratings are predicted, and top-n POIs are recommended which are more probably to be accepted by the target user.

3.3.1 Ant colony optimization

The new swarm intelligent algorithm called Ant colony optimization (ACO) [59] was introduced to address the NP-hard optimization problem. It inherits the foraging characteristics of an ant colony to obtain the optimal solution in high dimensional search space. Ants try to find their shortest way to their habitat from the food source through pheromone as a communicating factor. While moving, each ant deposits some amount of pheromone on its solution component and finds the next solution component with some probabilistic rules. The key feature of the ant colony optimization is the combination of posterior information about the last obtained local optima and prior information about the global optima. ACO is represented by the weighted graph, and each ant generates some amount of pheromone on the edges to communicate with each other while finding their optimal path. The amount of pheromone released indicates the quality of the solution obtained. To avoid local optima, the pheromone evaporates after some time which helps to explore a new path in the solution space. It significantly reduces the probability of choosing a longer path, as ants used to prefer the shortest path that contains a higher amount of pheromone. Considering the better exploration and exploitation feature of the ACO, it is utilized for user clustering.

3.3.2 Cuckoo search algorithm

Cuckoo Search Algorithm (CSA) [60] follows the biological features of cuckoo to solve complex optimization problems. CSA inherits the parasitic characteristics of cuckoo which lays its egg randomly on other birds' nest at a maximum distance. The dissimilar cuckoos' eggs are identified by the host bird and excluded from the nest; else the chick grows in the host nest. Otherwise, the host bird disposes its own nest and builds the new one somewhere else. Cuckoo used to move from one place to another in search of another nest. The breeding and transition behavior of the cuckoo is employed in data clustering to address the optimization problem of complex datasets. The key advantage of CSA is its simplicity and easy ignorance of local optima. But its convergence rate is slow and resulting in low clustering accuracy. To enrich the convergence rate and clustering accuracy, the nest finding distance is to be set flexible for global optimization. To determine the best global nest, the Markov chain method is utilized. In the Markov chain model, the presence of the next nearby nest is determined by the location of the current nest. The standard CSA may result in premature convergence, as all cuckoos follow the same searching method and got trap in the local optima. Thus, the combination of k-means with cuckoo search algorithm helps to enrich the convergence rate.

3.3.3 Mussels wandering optimization

The new optimization algorithm is developed as Mussels Wandering Optimization (MWO) inspired from the mussels' bed formation behavior at complex marine surfaces to form their habitat. The biological behaviors of mussels are inherited in MWO to obtain an optimal solution in clustering problems [40]. Mussels join together for physical protection and move less to maintain the bed density. However, if the cluster grows greater, mussels decide to move away and forms bed based on Random Levy walk. Let us assume that the mussel population consists of N individuals and forms the bed in the marine region of d-dimensional space. Let $f(s)$ be the objective function that represents nutrition. The MWO algorithm consists of the following six steps to provide optimal solutions. They are 1) initializing mussels population and other parameters, 2) computing both short and long range mussel density, 3) establishing the transition strategy, 4) updating the new position, 5) evaluating the fitness of updated position and 6) examining the terminate criteria. Notably, the bed formation with Random Levy walk to discover the most appropriate place for habitat represents the global optimization. Thus, the optimization algorithm developed from the ecological behavior of the mussels helps to provide an optimal solution.

3.4 Rating enhancement by co-training

In most of the real-time applications, data are described either by multiple distinct features or by multiple views of a single feature. Co-training is related to the problem of multi-view learning of the same dataset. In general, Co-training technique is implemented on datasets which consist of very few labeled data, and all others are unlabeled data. The co-training technique first learns the classifier for each view of labeled data and predictions are made on unlabelled data to obtain additional information. The obtained additional information is feed to other classifiers as input and process repeat for some iteration to enrich the classification accuracy. To provide additional information, each data set should be represented by at least two views. However, co-training can also be performed for a single representation of the data. To handle rating sparsity and cold start issues of the conventional recommender system, co-training method is exploited in the proposed recommendation generation process.

HLTRS uses single dataset as an input and three different swarm intelligent algorithms as multiple views. The similarity matrix produced by each clustering algorithm is combined to generate a consensus matrix for rating prediction. While co-training the clustering algorithms, each algorithm is trained independently to agree with the clusters of other algorithms, so that they learn not only from its own approach but also from other algorithms. At each step of the co-training module, the swarm intelligent based user clustering algorithm generates the similarity matrix, and it is added to the training dataset of other clustering algorithm and repeats the process iteratively. This additional information is exchanged over all other user clustering algorithms iteratively to enrich the rating matrix. In addition, the results of each clustering algorithm are combined to generate a consensus rating matrix for rating prediction and precise recommendation generation.

In this module, the HLTRS divides the original input dataset into the rated dataset r_u^l and unrated dataset u_u^l . In the rated dataset, the user-POI matrix is filled with ratings i.e. all the entries of user-POI rating matrix is filled, while unrated dataset consists of randomly selected users and POI. Since user-POI rating matrix is sparse and unrated dataset is large, including all the unrated user and POI into the prediction algorithm resulting in slow convergence. To reduce the burden of the co-training process, x number of user and POIs per user is randomly selected. The goal of the co-training module is to predict the rating for unvisited POI. The user similarities are computed to predict the ratings for unfamiliar POIs based on similar users' rating profiles. In HLTRS, Pearson coefficient is applied to estimate the similarity score and it is defined as follows:

$$sim(u_1, u_2) = \frac{\sum_{l \in S_{u_1,u_2}} \left(r_{u_1}^l - \overline{r}_{u_1}\right)\left(r_{u_2}^l - \overline{r}_{u_2}\right)}{\sqrt{\sum_{l \in S_{u_1,u_2}} \left(r_{u_1}^l - \overline{r}_{u_1}\right)^2} \sqrt{\sum_{l \in S_{u_1,u_2}} \left(r_{u_2}^l - \overline{r}_{u_2}\right)^2}} \tag{3}$$

where S_{u_1,u_2} is the set of POI visited and reviewed by both the users u_1 and u_2, $r_{u_1}^l$ is the POI l rated by the user u_1 and $\overline{r}_{u_1}^l$ is the average of all POI ratings provided by u_1. Similarly, $r_{u_2}^l$ is the POI l rated by u_2 and $\overline{r}_{u_2}^l$ is the average of all POI ratings of u_2. Then the rating prediction for unfamiliar POI of the user u_1 is given as follows

$$P_{u_1,l} = \overline{r}_{u_1} + \frac{\sum_{u_i \in k_{u_1,l}} sim(u_1, u_i)\left(r_{u_i,l} - \overline{r}_{u_i}\right)}{\sum_{u_i \in k_{u_1,l}} sim(u_1, u_i)} \tag{4}$$

where $\overline{r}_{u_1}$ is the average of all POI ratings given by u_1, $k_{u_1,l}$ is the k-nearest neighbors of the user u_1 who already rated the POI for which the rating is going to predicted on behalf of the user u_1, $r_{u_i,l}$ is the rate of the POI l given by the user u_i and $sim(u_1, u_i)$ is the similarity measurement between the user u_1 and u_i. After predicting the ratings for unvisited POI of the use u_1, the top-n rated POIs are suggested to the user as personalized recommendations. If the target user is not convinced with the recommended POIs, then the list is revised again and generated based on his requirements and preferences.

3.5 Incorporating contextual information for better travel recommendations

Travel Recommender system utilizes the experiences and opinions of the similar user of a community to recommend the point of interest from the set of available choices. The proposed HLTRS integrates the contextual information with user profiles to maximize performance efficiency. The contextual data such as season, social ties, weather, time and dynamic feature of target user i.e. emotions are considered as an important feature for travel recommender system. According to Adomavicius and Tuzhilin [61], contextual information can be incorporated either before or after applying the recommendation algorithm to filter the relevant items from the master set of choices. In the contextual pre-filtering model, contextual data are first utilized to filter irrelevant items and then the recommendation algorithm is applied for recommendation generation. But in the post-filtering method, the recommendation algorithm is first applied to the raw dataset, and then the contextual information is utilized to update the recommended list based on user preferences and requirements.

The proposed HLTRS incorporates a contextual pre-filtering method to filter out irrelevant information resulting in reduced computational complexity. The distance between the users' current location and POI is estimated using geographical features such as latitude and longitude. Then the list of point of interests located within the predefined radius is suggested to the target user based on their interests. Collaborative filtering algorithm builds the user-POI matrix from user profiles where each entry represents the ratings provided by the target user for the corresponding POI. Let us consider the rating matrix with m users, where $U = \{u_1, u_2, \cdots, u_m\}$ and n POI, where $L = \{l_1, l_2, \cdots, l_n\}$. Each user would have experienced some set of POIs, and they would have expressed their feedback as a 5-star rating. By computing the similarity between users, collaborative filtering algorithm identifies the neighbor set of same interest and predicts the rating for the unfamiliar POIs of the user. Then the list of top-n ranked location is recommended as the point of interest.

4 Experiments and discussions

The proposed Hybrid Location-based Travel Recommender System (HLTRS) is experimentally evaluated to demonstrate its efficiency and performance in generating a personalized list of Point of Interests. We conducted the experiment on real time complex TripAdvisor dataset for two different final Number of Recommendations (N_R) as 10 and 20. TripAdvisor is a popular travel recommendation web portal that comprises of users feedbacks and reviews on the venues and locations. The dataset used for the experimentation comprises of 9149 venues, 13,410 users, and 152,721 user ratings for locations. The dataset is divided into 80% for training and 20% for testing purpose. The results obtained from experimental study are compared with the other existing baseline methods such as User-based KNN [26], Item-based KNN [62], Biased MF [63], KBTRS [64], DPSOHiK [47], PCAHTRS [50], and HSSRS [10], and the analyses are presented for the better understanding of the effectiveness of HLTRS. Experiments are performed on the PC running on the 64-bit Windows 10 Operating System with Intel Core i7-5500U clocked at 3.00 GHz and 16 GB of memory. The experiments for generating recommendations on the mobile framework are conducted on the smartphone with Snapdragon 660 Octa-Core clocked at 1.95 GHz with 6 GB of memory.

4.1 Evaluation metrics

The major aim of conducting the experiments is to determine the ability of the proposed HLTRS for generating recommendation of the personalized POIs list to the target user. We exploit six standard evaluation metrics such as Mean Absolute Error (MSE), Root Mean Squared Error (RMSE), Coverage, Recall, Precision, and F-Measure to validate the performance efficiency of the recommendation approaches. The evaluation metrics are defined as follows.

4.1.1 Root mean squared error

Root Mean Squared Error is a popular recommendation evaluation metric used to estimate the accuracy of the predicted ratings, and it is calculated as follows.

$$RMSE = \sqrt{\frac{\sum_{user,poi}\left(ActualRatings_{user,poi} - PredictedRatings_{user,poi}\right)^2}{N}} \quad (5)$$

4.1.2 Mean absolute error

Mean Absolute Error is the alternative metric to RMSE used to compute the difference between the actual and predicted ratings, and it is computed as follows.

$$MAE = \frac{1}{N_{users}} \sum_{user=1}^{N_{users}} \left| Actual_{ratings}(poi_{user}) - Predicted_{ratings}(poi_{user}) \right| \quad (6)$$

4.1.3 Coverage

Coverage metric is used to determine the percentage of predicted ratings among the total number of ratings considered for the experiments. Coverage metric is defined as follows.

$$Coverage = \frac{Number\ of\ Predicted\ Ratings}{Total\ Number\ of\ Ratings\ in\ Test\ Set} \quad (7)$$

4.1.4 Precision

Precision is a popular evaluation metric used to determine the prediction and ranking ability of the recommendation approach by exploiting the final recommendations presented to the active target user. Precision metric considers the number of relevant POIs in the presented recommendation list to compute the positive prediction score and it is calculated as follows.

$$Precision = \frac{|RecommendedPOI(user) \cap RelevantPOI(user)|}{|RecommendedPOI(user)|} \quad (8)$$

4.1.5 Recall

Recall is also commonly known as the sensitive metric which is used to compute the usefulness of generated recommendations and it is calculated as follows.

$$Recall = \frac{|RecommendedPOI(user) \cap RelevantPOI(user)|}{|RelevantPOI(user)|} \quad (9)$$

4.1.6 F-measure

F-Measure is the harmonic mean of recall and precision metrics which is used to determine the quality of generated POI recommendations and it is defined as follows:

$$F\text{-}Measure = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (10)$$

4.2 Comparison of experimental results

This subsection presents the experimental results and analyses with standard evaluation metrics for the production of personalized travel recommendations on real time complex TripAdvisor dataset. From the info-graphics Figs. 3, 4, 5, 6, 7 and 8, it is apparent that our proposed HLTRS is proficient of producing better recommendation in terms of prediction and raking of recommendations. Recommendations are generated for the final number of recommendations (N_R) as 10 and 20. With focused designing of HLTRS to produce efficient and accurate recommendations, the proposed HLTRS has less RMSE and MAE compared to the existing baselines. On the coverage estimation, the HLTRS is capable of predicting maximum user-item pairs in the dataset used for the experimentation. While considering the precision and recall, the HLTRS has performed well for the different number of recommendations. On evaluating the quality of the recommendations, the HLTRS has highest F-Measure score over existing standalone and hybrid approaches. The experimental result reveals the enhanced performance of the hybrid approaches over standalone models. In terms of coverage, User-based KNN and Item-based KNN have the least capacity to make predictions to produce recommendations, which makes these approaches to produce poor quality recommendations. The recommendation models of Biased MF and KBTRS have higher error rate on the RMSE and MAE which makes the final recommendation list inconsistent. The recommendation approaches of DPSOHiK, PCAHTRS, and HSSRS have produced comparative results to the proposed HLTRS.

The efficient utilization of user profiling, context-aware data, and swarm intelligence algorithms enhance the rating prediction model of the proposed HLTRS, and the proposed approach is capable of producing recommendations to both cold start, and all user sets. The dataset used for experimentation is large and the time taken to produce the final set of

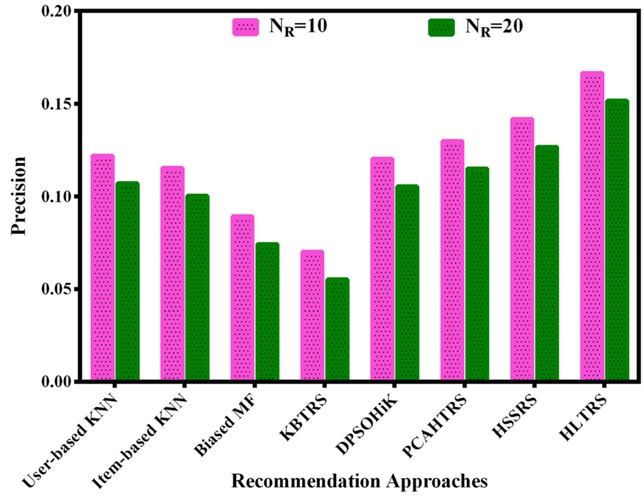


Fig. 3 Comparison of precision for various recommendation approaches

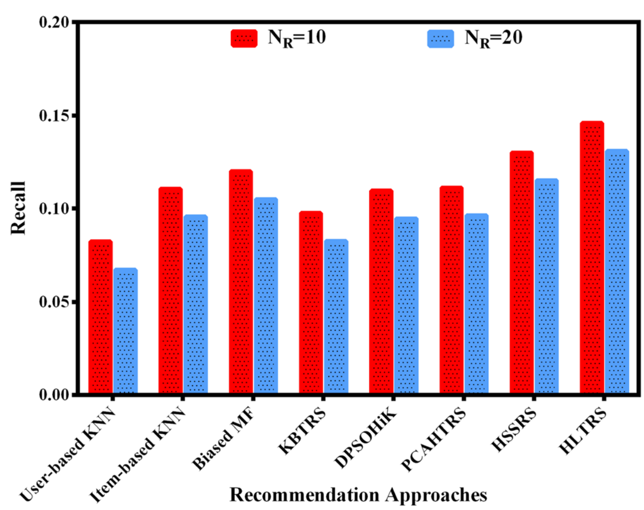


Fig. 4 Comparison of recall for various recommendation approaches

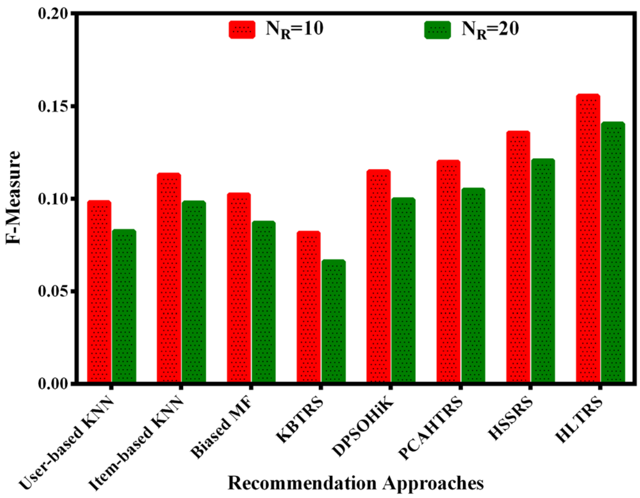


Fig. 5 Comparison of F-Measure for various recommendation approaches

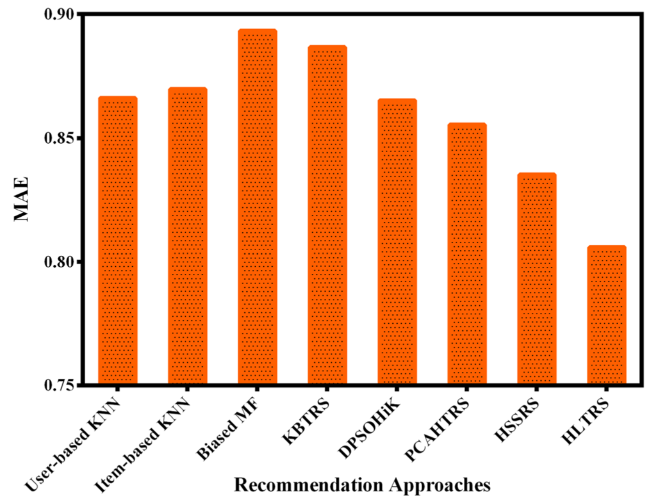


Fig. 7 Comparison of MAE for various recommendation approaches

recommendations to an active user is also considered to be a crucial factor. With the complex recommendation generation process of HLTRS, the proposed approach has made a positive impact with reliable recommendations generated in the optimal computational cost. After determining the relevant set of POIs, to generate the final list of recommendations, the proposed HLTRS employs conditional filtering approach by exploiting the user's current location, contextual data, and travel distance preference. The conditional filtering approach helps the HLTRS to produce better POI recommendations to the active user. Based on the target users' current location and travel distance preferences, the POI recommendation list is re-ranked and the final top-n recommendations are generated through the user interface. The sparsity of the ratings data makes a direct impact on the recommendation generation, and scalability of the recommendation model also considered as an essential feature. By estimating the leaning time taken by the proposed approach, we found that leaning time raises with the increasing size of the dataset used. From the experimental results, it has been demonstrated that the recommendation approaches produce accurate recommendation when the final number of recommendations are less and when the number of the final top-n list increases it crunches the accuracy of the recommendation list.

From the experiments conducted and the analyses made on the obtained results, the following conclusions are derived.

- The utilization of multiple swarm intelligence algorithms is helpful in improving the accuracy of the recommendations generated.
- The proposed HLTRS with utilization of user profiling, context-aware data and trust information has demonstrated its ability to produce reliable POI recommendations.
- The performance of the HLTRS has been improved with the co-training based rating enhancement approach.

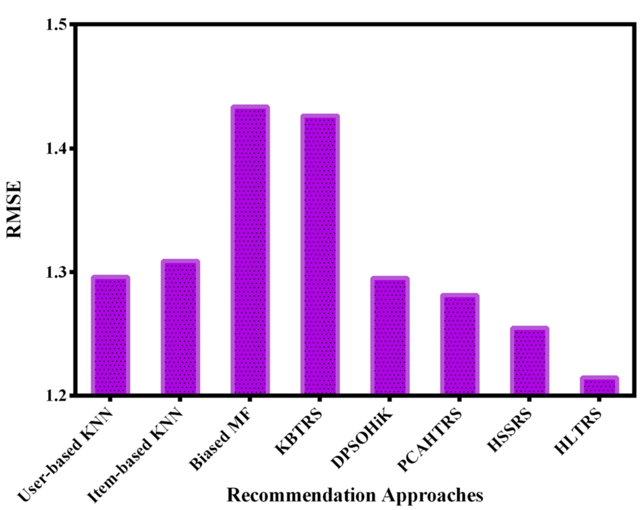


Fig. 6 Comparison of RMSE for various recommendation approaches

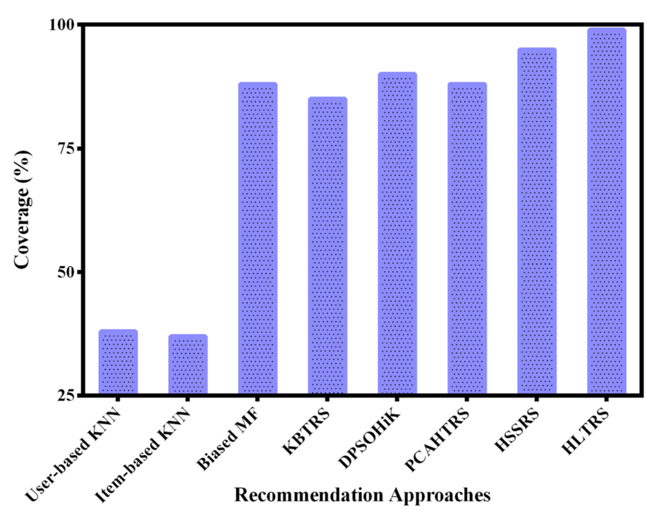


Fig. 8 Comparison of coverage for various recommendation approaches

- HLTRS outperforms standalone approaches and other baseline hybrid approaches with the incorporation of user's contextual information.
- The proposed HLTRS has also been executed on the high-performance computing clusters for the generation of top-n POI list and travel planning.
- HLTRS is evaluated for the real-time recommendation scenarios on the mobile environment, and the suggested recommendations are user satisfiable and proficient for travel requirements.
- Useful and new travel recommender system has been developed and presented for the generation of better recommendations to meet the changing travel requirements of the active target user.

4.3 Recommendations on mobile and evaluation

In the travel context, smartphones help users in assisting them during the navigational difficulties faced while searching for information on a venue or travel route. The travel recommender systems on the mobile platform help users by providing information on the relevant POIs and available e-services such as trip planning and mapping services. The mobile-based travel recommender systems are found to cost-effective and user satisfiable over the conventional web-based solutions. As the travel planning deals with the more than one POI, modeling the user profile and generating the personalized trip with the changing contextual information is a complex task. The personalized tour plan has to be designed based on target user's constraints such as preferable distance, travel mode and feasibility, energy level, accessible location categories, etc. The target user's constraints with preferences and features of the POIs are mapped to solve the travel planning problem.

The proposed HLTRS is evaluated on the real time scenarios by exploiting mobile recommendation frameworks such as XplorerVU [58], XplorerVU TwB [10], and CHXplorer [12]. The mobile recommendation frameworks of XplorerVU, XplorerVU TwB, and CHXplorer are presented in Fig. 9. The proposed HLTRS on the mobile recommendation framework generates personalized recommendations based on target users' interests and preferences. The users interact with the recommendation engine through mobile application's user interface. The co-trained swarm intelligent algorithms are utilized on the mobile recommendation framework to produce personalized POIs and feasible travel plan. The POIs with higher predicted ratings are re-ranked on the basis of the user's contextual constraints, and the final set of top-n point of interests are suggested as recommendations to the target user. To reduce the sparsity and cold start difficulty of the travel recommender system, HLTRS exploits the location-based social network data to acquire users' current location, historical check-ins, preferences, contextual information, interests, and requirements. Based on the input provided, the recommendation set is further refined and the modified recommendations are generated and provided to the user through the mobile interface of recommendation framework.

The mobile user interface presents the generated personalized POI list with the interactive map feature to provide enhanced user personalization. All three mobile recommendation frameworks provide a "save the plan" option. And these saved trips and plans are considered as user's explicit preferences and will be exploited for the future recommendation generation. The XplorerVU is designed for individual users, XplorerVU TwB is built for a group of users, and CHXplorer is developed for recommending cultural heritage locations. The mobile recommendation frameworks have been already proven to be user-friendly with varying users knowledge levels from the common man to experts. With the user-friendly travel planning process of the proposed HLTRS and proven mobile recommendation frameworks, the travel recommender system produced single destination recommendation, multiple POI recommendation, and travel trip plans. The user interfaces of mobile recommendation frameworks are evaluated on the ISO 9241-11-110 standards. The recommendation ability of HLTRS on the mobile user interface is evaluated on three mobile recommendation frameworks with 86 participants. With extensive user study on three proven mobile recommendation frameworks, HLTRS has demonstrated its improved performance by providing travel planning and recommendations by incorporating the implicit and explicit preferences of the active target users.

4.4 Location-based recommendations for cultural heritage applications

The rapid development of internet technologies and smartphone applications have reflected with a massive increase in the variety of location-based services provided to digital users. With the better utilization of emerging intelligent technologies, recommender systems are employed to support the travel users visiting the cultural heritage sites. The cultural heritage recommendations differ from the traditional travel location recommendation as the features and attributes of cultural heritages venues requires specific mapping to the users' interests. With the rich information management system specifically designed and developed for the cultural heritage locations, CHXplorer has demonstrated its ability to be an effective mobile decision support mechanism for the cultural heritage travellers. We have employed our proposed HLTRS on the CHXplorer mobile recommendation framework to generate tailored cultural heritage recommendations. The experiments are conducted at the cultural heritage sites of the Thanjavur city in India for the prediction of the next cultural heritage location and travel plan generation. The experimental results and the user study made reveal the enhanced

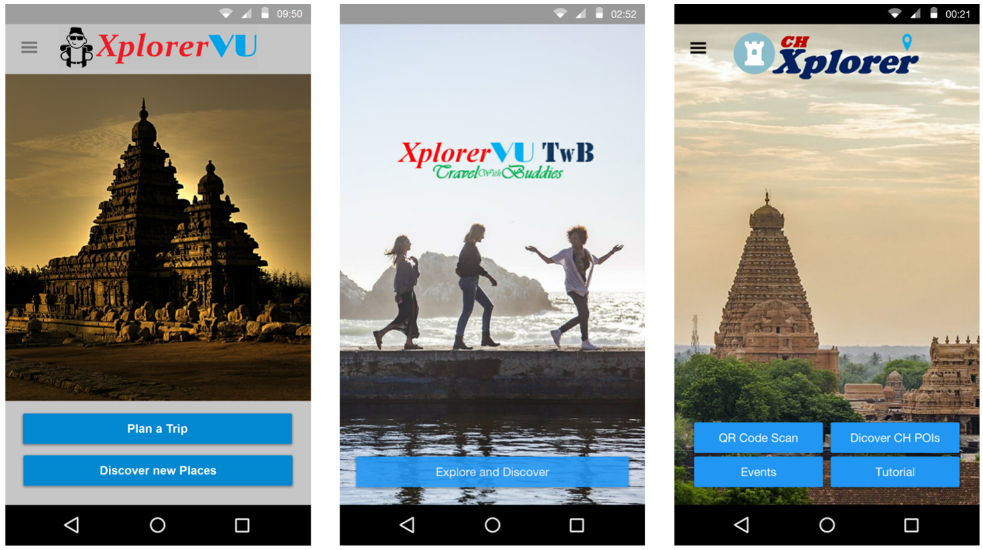


Fig. 9 Mobile recommendation frameworks **a** XplorerVU [58] **b** XplorerVU TwB [10] **c** CHXplorer [12]

performance of HLTRS on CHXplorer and positive user feedback on the generated personalized recommendations. Along with the travel recommendations to the cultural heritage locations, the CHXplorer also presents the multimedia content on the user interface with backend support from the dedicated information management system. The mobile interface of the CHXplorer is devised to be interactive and user-friendly for all types of users and based on user's interactions; recommendation engine is capable of discovering the user's implicit preferences. The archaeological cultural heritage sites in the Thanjavur, India such as Brihadeeshwara Temple, Schwartz Church, and Thanjavur Fort Place were considered for the experimentation of HLTRS on CHXplorer mobile recommendation framework to evaluate the developed model in real time scenarios. The obtained results revealed the promising performance of HLTRS for the focused recommendations for cultural heritage travelers with enhanced prediction model through exploiting the preferences of the active target user.

5 Conclusion and future directions

The significant progress in the development of recommender systems has proved their efficiency in handling information overloading problem in various application domains. To solve the user interest prediction and personalization problems of the travel recommender system, we have developed a new recommendation model as Hybrid Location-based Travel Recommender System (HLTRS) to generate the tailor-made point of interest recommendations by considering target user's dynamic needs and preferences. To generate personalized recommendations, HLTRS constructs an individual user profile for predicting the interesting locations based on the individual and group activity information obtained from the location-based social network. The constructed individual user profile comprises of the explicit and implicit preferences of the target user, and it is used in the recommendation generation process. The developed HLTRS utilizes trust and contextual information of the target user to filter highly relevant POI from the master item set resulting in reduced computational complexity. The ensemble-based co-trained swarm intelligence approach of HLTFS reflects with the enrichment of the user-POI rating matrix. The developed HLTRS is designed with the focus to solve the sparsity and cold start problems of the conventional recommender system. The proposed HLTRS is experimentally evaluated on real time dataset of TripAdvisor for the production of personalized travel recommendations. We have also conducted extensive user study for estimating the capacity of HLTRS for producing personalized travel recommendations in real time scenarios. The obtained experimental results demonstrate the improved performance of the HLTRS over the existing baselines and user study reveals promising recommendation ability of HLTRS on various mobile recommendation frameworks. In the future, we plan to examine the recommender systems research for the cross-domain recommendations, and also we intend to study the consensus decision-making problem in the group travel recommendations.

Acknowledgements The authors are grateful to the 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 Deemed University, Thanjavur, for providing the infrastructural facilities to carry out this research work.

References

1. Bobadilla J, Ortega F, Hernando A, Gutiérrez A (2013) Recommender systems survey. Knowl-Based Syst 46:109–132
2. Leino J (2014) User factors in recommender systems: Case studies in e-commerce, news recommending, and e-learning

3. Subramaniyaswamy V, Vijayakumar V, Logesh R, Indragandhi V (2015) Intelligent travel recommendation system by mining attributes from community contributed photos. Procedia Computer Science 50:447–455
4. Subramaniyaswamy V, Logesh R, Chandrashekhar M, Challa A, Vijayakumar V (2017) A personalised movie recommendation system based on collaborative filtering. International Journal of High Performance Computing and Networking 10(1–2):54–63
5. Subramaniyaswamy V, Manogaran G, Logesh R, Vijayakumar V, Chilamkurti N, Malathi D, Senthilselvan N (2018) An ontology-driven personalized food recommendation in IoT-based healthcare system. J Supercomput:1–33. https://doi.org/10.1007/s11227-018-2331-8
6. Subramaniyaswamy V, Logesh R, Indragandhi V (2018) Intelligent sports commentary recommendation system for individual cricket players. International Journal of Advanced Intelligence Paradigms 10(1–2):103–117
7. Vairavasundaram S, Varadharajan V, Vairavasundaram I, Ravi L (2015) Data mining-based tag recommendation system: an overview. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 5(3):87–112
8. Zhang Y, Tu Z, Wang Q (2017) TempoRec: Temporal-topic based recommender for social network services. Mobile Networks and Applications 22(6):1182–1191
9. Logesh R, Subramaniyaswamy V, Vijayakumar V (2018) A personalised travel recommender system utilising social network profile and accurate GPS data. Electronic Government, an International Journal 14(1):90–113
10. Logesh R, Subramaniyaswamy V, Vijayakumar V, Li X (2018) Efficient User Profiling Based Intelligent Travel Recommender System for Individual and Group of Users. Mobile Networks and Applications:1–16. https://doi.org/10.1007/s11036-018-1059-2
11. Logesh R, Subramaniyaswamy V (2017) Learning Recency and Inferring Associations in Location Based Social Network for Emotion Induced Point-of-Interest Recommendation. J Inf Sci Eng 33(6)
12. Ravi L, Vairavasundaram S, Palani S, Devarajan M (2019) Location-based personalized recommender system in the internet of cultural things. J Intell Fuzzy Syst. https://doi.org/10.3233/JIFS-169973
13. Logesh R, Subramaniyaswamy V (2017) A Reliable Point of Interest Recommendation based on Trust Relevancy between Users. Wirel Pers Commun 97(2):2751–2780
14. Chen X, Xu X, Huang JZ, Ye Y (2013) TW-k-means: Automated two-level variable weighting clustering algorithm for multiview data. IEEE Trans Knowl Data Eng 25(4):932–944
15. Logesh R, Subramaniyaswamy V, Vijayakumar V, Gao XZ, Wang GG (2019) Hybrid bio-inspired user clustering for the generation of diversified recommendations. Neural Comput & Applic. https://doi.org/10.1007/s00521-019-04128-6
16. Wang GG, Gandomi AH, Alavi AH, Deb S (2016) A hybrid method based on krill herd and quantum-behaved particle swarm optimization. Neural Comput & Applic 27(4):989–1006
17. Balusamy B, Karthikeyan K, Sangaiah AK (2017) Ant colony-based load balancing and fault recovery for cloud computing environment. International Journal of Advanced Intelligence Paradigms 9(2–3):204–219
18. Wang GG (2018) Moth search algorithm: a bio-inspired metaheuristic algorithm for global optimization problems. Memetic Computing 10:151–164. https://doi.org/10.1007/s12293-016-0212-3
19. Devarajan M, Fatima NS, Vairavasundaram S, Ravi L (2019) Swarm intelligence clustering ensemble based point of interest recommendation for social cyber-physical systems. J Intell Fuzzy Syst. https://doi.org/10.3233/JIFS-169991
20. Guo G (2013) Improving the performance of recommender systems by alleviating the data sparsity and cold start problems. In Twenty-Third International Joint Conference on Artificial Intelligence
21. Da Costa AF, Manzato MG, Campello RJ (2018) CoRec: a co-training approach for recommender systems. In: Proceedings of the 33rd Annual ACM Symposium on Applied Computing (pp. 696–703). ACM
22. Blum A, Mitchell T (1998) Combining labeled and unlabeled data with co-training. In: Proceedings of the eleventh annual conference on Computational learning theory (pp. 92–100). ACM
23. da Costa AF, Manzato MG, Campello RJ (2019) Boosting collaborative filtering with an ensemble of co-trained recommenders. Expert Syst Appl 115:427–441
24. Ravi L, Vairavasundaram S (2016) A collaborative location based travel recommendation system through enhanced rating prediction for the group of users. Computational Intelligence and Neuroscience 2016:7. https://doi.org/10.1155/2016/1291358
25. Goldberg D, Nichols D, Oki BM, Terry D (1992) Using collaborative filtering to weave an information tapestry. Commun ACM 35(12):61–71
26. Konstan JA, Miller BN, Maltz D, Herlocker JL, Gordon LR, Riedl J (1997) GroupLens: applying collaborative filtering to Usenet news. Commun ACM 40(3):77–88
27. Herlocker JL, Konstan JA, Terveen LG, Riedl JT (2004) Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems (TOIS) 22(1):5–53
28. Sankar H, Subramaniyaswamy V, Vijayakumar V, Kumar SA, Logesh R, Umamakeswari A (2019) Intelligent sentiment analysis approach using edge computing-based deep learning technique. Software: Practice and Experience. https://doi.org/10.1002/spe.2687
29. Subramaniyaswamy V, Logesh R, Abejith M, Umasankar S, Umamakeswari A (2017) Sentiment Analysis of Tweets for Estimating Criticality and Security of Events. Journal of Organizational and End User Computing (JOEUC) 29(4):51–71
30. Malathi D, Logesh R, Subramaniyaswamy V, Vijayakumar V, Sangaiah AK (2019) Hybrid Reasoning-based Privacy-Aware Disease Prediction Support System. Comput Electr Eng 73:114–127
31. Lathia N, Hailes S, Capra L (2008) Trust-based collaborative filtering. In: IFIP international conference on trust management (pp. 119–134). Springer, Boston
32. Massa P, Avesani P (2009) Trust metrics in recommender systems. In: Computing with social trust (pp. 259–285). Springer, London
33. Kant V, Bharadwaj KK (2013) Fuzzy computational models of trust and distrust for enhanced recommendations. Int J Intell Syst 28(4):332–365
34. Guo G, Zhang J, Thalmann D, Basu A, Yorke-Smith N (2014) From ratings to trust: an empirical study of implicit trust in recommender systems. In: Proceedings of the 29th Annual ACM Symposium on Applied Computing (pp. 248–253). ACM
35. Gupta S, Nagpal S (2015) Trust aware recommender systems: a survey on implicit trust generation techniques. International Journal of Computer Science and Information Technologies 6(4):3594–3599
36. Ujjin S, Bentley PJ (2003) Particle swarm optimization recommender system. In: Proceedings of the 2003 IEEE Swarm Intelligence Symposium. SIS'03 (Cat. No. 03EX706) (pp. 124–131). IEEE
37. Katarya R, Verma OP (2016) A collaborative recommender system enhanced with particle swarm optimization technique. Multimed Tools Appl 75(15):9225–9239
38. Choudhary V, Mullick D, Nagpal S (2017) Gravitational search algorithm in recommendation systems. In International Conference on Swarm Intelligence (pp. 597–607). Springer, Cham

39. Bedi P, Sharma R (2012) Trust based recommender system using ant colony for trust computation. Expert Syst Appl 39(1):1183–1190
40. An J, Kang Q, Wang L, Wu Q (2013) Mussels wandering optimization: an ecologically inspired algorithm for global optimization. Cogn Comput 5(2):188–199
41. Bouyer A, Hatamlou A (2018) An efficient hybrid clustering method based on improved cuckoo optimization and modified particle swarm optimization algorithms. Appl Soft Comput 67:172–182
42. Devarajan M, Ravi L (2018) Intelligent cyber-physical system for an efficient detection of Parkinson disease using fog computing. Multimed Tools Appl. https://doi.org/10.1007/s11042-018-6898-0
43. Vijayakumar V, Malathi D, Subramaniyaswamy V, Saravanan P, Logesh R (2018) Fog computing-based intelligent healthcare system for the detection and prevention of mosquito-borne diseases. Comput Hum Behav. https://doi.org/10.1016/j.chb.2018.12.009
44. Gavalas D, Kasapakis V, Konstantopoulos C, Pantziou G, Vathis N, Zaroliagis C (2015) The eCOMPASS multimodal tourist tour planner. Expert Syst Appl 42(21):7303–7316
45. Cenamor I, de la Rosa T, Núñez S, Borrajo D (2017) Planning for tourism routes using social networks. Expert Syst Appl 69:1–9
46. De Pessemier T, Dhondt J, Martens L (2017) Hybrid group recommendations for a travel service. Multimed Tools Appl 76(2):2787–2811
47. Logesh R, Subramaniyaswamy V, Malathi D, Senthilselvan N, Sasikumar A, Saravanan P (2017) Dynamic particle swarm optimization for personalized recommender system based on electroencephalography feedback. Biomed Res 28(13):5646–5650
48. Brilhante IR, Macedo JA, Nardini FM, Perego R, Renso C (2015) On planning sightseeing tours with TripBuilder. Inf Process Manag 51(2):1–15
49. Kurata Y, Hara T (2013) CT-planner4: Toward a more user-friendly interactive day-tour planner. In: Information and communication technologies in tourism 2014 (pp. 73–86). Springer, Cham
50. Logesh R, Subramaniyaswamy V (2019) Exploring Hybrid Recommender Systems for Personalized Travel Applications. In: Cognitive Informatics and Soft Computing (pp. 535–544). Springer, Singapore
51. Subramaniyaswamy V, Logesh R (2017) Adaptive KNN based recommender system through Mining of User Preferences. Wirel Pers Commun 97(2):2229–2247
52. Zhang M, Tang J, Zhang X, Xue X (2014) Addressing cold start in recommender systems: A semi-supervised co-training algorithm. In: Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval (pp. 73–82). ACM
53. Zhang Q, Wang H (2015) Collaborative Multi-view Learning with Active Discriminative Prior for Recommendation. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining (pp. 355–368). Springer, Cham
54. Quang TN, Phuong ND (2015) Collaborative filtering by co-training method. In: Knowledge and Systems Engineering (pp. 273–285). Springer, Cham
55. Matuszyk P, Spiliopoulou M (2017) Stream-based semi-supervised learning for recommender systems. Mach Learn 106(6):771–798
56. Logesh R, Subramaniyaswamy V, Malathi D, Sivaramakrishnan N, Vijayakumar V (2019) Enhancing Recommendation Stability of Collaborative Filtering Recommender System through Bio-inspired Clustering Ensemble method. Neural Comput & Applic. https://doi.org/10.1007/s00521-018-3891-5
57. Younus A, O'Riordan C, Pasi G (2014) A language modeling approach to personalized search based on users' microblog behavior. In: European Conference on Information Retrieval (pp. 727–732). Springer, Cham
58. Logesh R, Subramaniyaswamy V, Vijayakumar V, Gao XZ, Indragandhi V (2018) A hybrid quantum-induced swarm intelligence clustering for the urban trip recommendation in smart city. Futur Gener Comput Syst 83:653–673
59. Dorigo M, Birattari M, Stützle T (2006) Ant Colony Optimization-Artificial Ants as a Computational Intelligence Technique. IEEE Computational Intelligence Magazine
60. Yang XS, Deb S (2009) Cuckoo search via Lévy flights. Nature & Biologically Inspired Computing. In 2009 NaBIC 2009 World Congress on: IEEE (pp. 210–214)
61. Adomavicius, G., & Tuzhilin, A. (2011). Context-aware recommender systems. In: Recommender systems handbook (pp. 217–253). Springer, Boston
62. Deshpande M, Karypis G (2004) Item-based top-n recommendation algorithms. ACM Transactions on Information Systems (TOIS) 22(1):143–177
63. Koren Y, Bell R, Volinsky C (2009) Matrix factorization techniques for recommender systems. Computer (8):30–37
64. Vijayakumar V, Vairavasundaram S, Logesh R, Sivapathi A (2019) Effective Knowledge Based Recommender System for Tailored Multiple Point of Interest Recommendation. International Journal of Web Portals (IJWP) 11(1):1–18