Neural Computing and Applications (2018) 30:1983–1990
https://doi.org/10.1007/s00521-017-3338-4

ORIGINAL ARTICLE

Movie recommender system with metaheuristic artificial bee

Rahul Katarya[1]

Received: 30 October 2017 / Accepted: 28 December 2017 / Published online: 6 January 2018
© The Natural Computing Applications Forum 2018

Abstract
Recommender systems are information retrieval tool that allocates accurate recommendations to the specific users. Collaborative movie recommender systems support users in accessing their popular movies by suggesting similar users or movies from their past common ratings. In this research work, a hybrid recommender system has been proposed which utilized k-means clustering algorithm with bio-inspired artificial bee colony (ABC) optimization technique and applied to the Movielens dataset. Our proposed system has been described systematic manner, and the subsequent results have been demonstrated. The proposed system (ABC-KM) is also compared with existing approaches, and the consequences have been examined. Estimation procedures such as precision, mean absolute error, recall, and accuracy for the movie recommender system delivered improved results for ABC-KM collaborative movie recommender system. The experiment outcomes on Movielens dataset established that the projected system provides immense achievement regarding scalability, performance and delivers accurate personalized movie recommendations by reducing cold start problem. As far as our best research knowledge, our proposed recommender system is novel and delivers effective fallouts when compared with already existing systems.

Keywords Recommender systems · Collaborative filtering · K-mean · Artificial bee colony

1 Introduction

Recommender systems are effective data filtering tools that are responsible for handling online data and information overload [1–5]. The determination of a recommender system is to create recommendations automatically of things for users preferences. Movie suggestion is the most broadly used interfaces united with web-based portals that intention is to support online library [6–8]. The mainstream of current systems is based on a collaborative filtering (CF) methodology that has been magnificently established [9–15]. It accumulates assessments of films given by an individual and then endorses specific films to the specific user. However, most of the time, recommender systems are suffered from integral restrictions such as reduced scalability and cold start complications [3, 16–19]. We have developed a hybrid collaborative movie recommender system with improved movie prediction accuracy. In our collaborative movie recommender system, we have employed k-mean and artificial bee colony (ABC) to design an effective ABC-KM-based movie recommender systems. The collaborative algorithms are based on similar users and the person who has similar tastes as their friends have. Present approaches for recommendation systems are either memory based or model based. Model-based techniques use a set of constraints, and these constraints can be reduced using basic reduction techniques. Model-based techniques handle the sparsity issues in a better way than memory-based technologies [20, 21]. User-based collaborative filtering is functional on Movielens and other e-commerce applications [22, 23]. ABC is the most recent optimization procedure that encourages detecting the smart foraging conduct of a honey bee swarm and that is why it is used in research work for optimization [24–27]. ABC is also a bio-inspired algorithm, and bio-inspired means the intelligent algorithms which are motivated by the biological working of bees and apply in optimization and computing [28–33]. There are numerous domains in which ABC has been applied for better results. The various domains are such as image processing, computer and

✉ Rahul Katarya
rahulkatarya@dtu.ac.in

[1] Department of Information Technology, Delhi Technological University, Delhi, India

sensor networks, power distribution, data clustering, industrial engineering, mechanical engineering, and protein structure. The purpose of this research work is to employ user-based collaborative filtering with ABC algorithm and obtain the results by computing similarity in a set of users with Pearson similarity on the set of users. Sometimes, user-item matrix with collaborative filtering can be enormous and leads to the sparse and cold start problems [34]. Collaborative filtering-based recommender systems are mostly affected by cold start problem, in which new users will need to rate an adequate number of items to permit the system to detect their choices accurately and then the only system will deliver trustworthy recommendations. This paper is arranged as Sect. 2 explains the survey work that was performed on collaborative recommender system and clustering-based collaborative endorsements. The suggested system is named as a k-mean-ABC movie recommender system and explained in Sect. 3. In Sect. 4, experiment fallouts performed on Movielens dataset are described and finally summarization of this article with and the upcoming work is highlighted in Sect. 5.

2 Related work

Recommender systems (RS) are most effective knowledge management systems that help users to filter unusable data and contribute to avoiding information overloading and deliver personalized ideas [35, 36]. Currently, CF is the furthermost operative procedure engaged by movie recommender systems, which is operated by the nearest-neighbor method [37–42]. There are two most popular techniques adopted by CF. Some authors improved memory-based procedures for recommender systems by determining the data sparsity issue in which they utilized support vector machine that computed likenesses between items and enhanced basic memory-based technique [43]. The deficiency of effective approaches for searching the right content may lead to a continuous damage of users, and the authors provided a study on this issue [44]. Computational intelligence has shown a significant impact in the field of data mining and e-commerce applications. In similar scenario, artificial immune system has shown significant influence on recommendation systems. Authors also proposed new mathematical expressions for analyzing immune network by utilizing Pearson correlation coefficient for Movielens and EachMovie datasets [45]. The k-mean algorithm has been widely used clustering approach in data mining [46, 47]. A hybrid movie recommender system was offered by utilizing k-means clustering and genetic algorithms (GAs) with principal component analysis (PCA) on Movielens dataset [7]. A competent incremental collaborative filtering recommender system was presented which was based on a weighted clustering method [48]. Computational intelligence also showed a significant role in various interdisciplinary environments like movie-based collaborative recommender systems. A collaborative filtering framework was proposed which assimilates both subjective and independent knowledge to produce recommendations for the user that solved the problem of sparsity and the cold start problem by using Movielens datasets [49]. ABC is the furthermost introduced optimization procedure that checks the brilliant expression of a honey bee swarm. We have utilized swarm optimization procedure in the proposed collaborative movie recommender system which showed the better and improved results when compared to the existing methods [24, 25].

3 k-means-artificial bee colony collaborative filtering framework

To overcome the limitations of a collaborative recommender system, we presented a hybrid cluster and optimization-based technique to advance movie prediction correctness. Our motive is to design a unified model solution that incorporates user ratings from the Movielens dataset for predictions. We used K-mean and ABC as optimization procedure and then apply to Movielens dataset for improved efficient recommender systems. Primarily, k-means clustering algorithm is applied to Movielens dataset for clustering of users into different clusters. The clusters are selected randomly at initial then users are checked one by one by calculating the differences in their ratings and the centroid of the clusters, and if their difference is lowest, then the user is assigned to the cluster to which they are nearby. However, at this instant not assure that each user has been assigned to the correct cluster with the lowest difference of centroid. So each user's distance is compared to cluster mean and displace the users according to the minimum distance from any cluster's mean. Now, this iterative repositioning would now continue from this new partition until no more rearrangements occur. After a point, if no more relocations occurred then that point is the point of completion of the clustering process. The k-means algorithm's necessary steps are presented in Fig. 1.

Next ABC algorithm is employed to the resultant of the k-means procedure for optimizing the results. The cluster is prepared but not optimized, so a function is developed, called as fitness function that helps in improving the user's centroid distances. Fitness function transforms the previous centroids for a limited number of iteration (i.e., relocation of centroids to users). Then, it classifies the users again by calculating the minimum centroid differences or applying k-means again. An artificial bee colony optimization

Fig. 1 Presentation of K-mean procedure

1. For clustering, Put K marks into the area characterized by the users. These marks symbolize a primary set of centroids.
2. Allocate individual user to the collection (cluster) that has the nearby centroid.
3. When totally users have been allocated, recomputed the locations of K centroids for the separate cluster.
4. Replicate stages 2 and 3 til no development in centroids. Group (clusters) is created.

algorithm may be described by using the following three idealized rules [24].

Initialize.
Repeat.
(a) Put the working bees on the nourishment bases in the memory;
(b) Put the observer bees on the nourishment bases in the memory;
(c) Dispatch the pathfinders to the hunt area for noticing new nourishment sources.
Till (necessities are met).

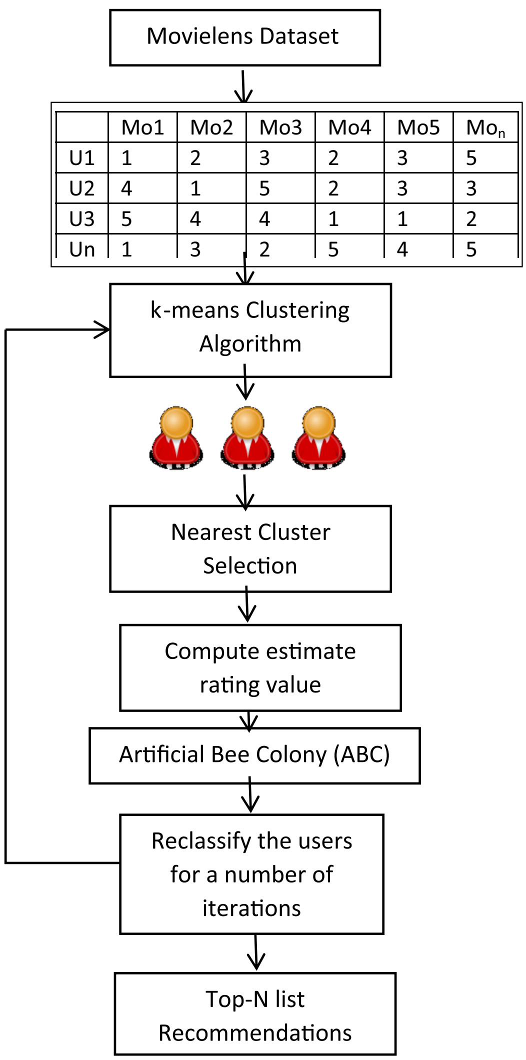

Fig. 2 Overview of proposed movie recommender system framework

Figure 2 shows a flowchart that displays the stepwise process that how artificial bee colony algorithm is applied. Figure 2 shows the above-described approach, which is being implemented to the Movielens dataset. The Movielens dataset is recorded by reading the file, and dataset is divided into clusters by using k-means clustering algorithm which results in k clusters so that each cluster has a centroid. The space between the user and the centroid is computed, and the user is placed in the cluster whose centroid is the least distance away from him. When all such users have been relocated, the centroids are repositioned, and the new positions are also calculated. Consequently, the estimated rating that the user will give is calculated, and framework is optimized using artificial bee colony algorithm. We have calculated various evaluation expressions for expecting the truth of recommender system such as precision, MAE, recall and accuracy that are acceptable and efficient when compared with already existing methodologies.

ABC determines a community of premilinary resolution vectors with superior results and neighbor exploration methodology, and then continuously proceeds by reducing an objective function (Fig. 3). The objective function $f\,(\vec{z})$ should be reduced by detecting vector $(\vec{z})$.

$$\text{minimize } f(\vec{z}),\ (\vec{z}) = (z1, z2, \ldots, zi, \ldots, zn-1, zn) \in Rn \tag{1}$$

which is controlled by the succeeding similarities and equivalences:

$$li \le zi \le ui, i = 1, \ldots, n \tag{2}$$

$$\text{substance as} : gy(\vec{z}) \le 0, \text{for } y = 1, \ldots, p \tag{3}$$

$$hy\,(\vec{z}) = 0, \text{for } y = p+1, \ldots, q \tag{4}$$

K-means clustering procedure relies on the preliminary locations and continuously congregates to the nearby local prime from the early point. Users require presetting the k value and centering points, which will often have an immense impact on cluster results. Hence, in the procedure of users gathering, we employed ABC procedure to conquer the K-mean clustering procedure. Therefore, in our proposed system, we employed ABC procedure to regulate the optimum value of center points and the resolutions compare with group's centers. The

```
Algorithm: ABC-KM Collaborative Filtering Pseudo Code
For each movie Mo1 in User U1's list
        For each user U2 who rated  Mo1
                For each movie Mo2 purchased by
                User U1
                Record that a user  rated  Mo1 and  Mo2
        For each movie  Mo2
        Compute the similarity between Mo1 and  Mo2
Cluster Initialization:
Reset the center of the clusters
Represent every close cluster as a data point
Compute mean of all data points and detect the location of the cluster.
Repeat above phases until convergence.
Population Initialisation:
 Prepare the communities zi,j , i = 1 ...SN,j = 1 ...D
Estimate  communities
 period=1
 reiteration
 Generte fresh results υi,j for the laboring bees by  Equation 2 and compute
 Execute the greedy choice procedure (GCP)
 Compute the probability by Equation 1.
 Compute fresh answers υi,j from the zi,j.
 Apply the GCP
 Regulate the unrestrained answer for the scout, if occurs, and interchange with a fresh result xi,j by (3)
 Learn the finest answer
 period = period +1
 until period=MPN(Maximum period Number)
```

Fig. 3 Pseudocode of proposed system

working of our proposed system with ABC and k-mean can be précised as follows: First, initialize randomly the locations of food bases (each food base being a set of centroids), employ the k-means algorithm to complete clustering task for all created locations, and compute the fitness value of particular group of centroids. In the next step, bees hunt fresh food bases and inform the place of food bases by working bees. We applied k-means clustering algorithm to estimate new fitness values and match them with the original ones. Enhanced food bases will be supplied to onlooker bees. Then, we determined the probability values of food sources and updated their place according to the probability values by onlooker bees. Again, the k-means algorithm was applied to finish clustering process, estimate new fitness values, and compare them with the original ones to update them. We checked the trial counter of food sources, formed a new nourishment foundation (set of centroids), and repeated the steps until the termination criterion is encountered. Therefore, Figs. 2 and 3 illustrate the user and movie behavior who adopted the same working flow as we explained above.

4 Experiment and results

We adopted the publically available Movielens dataset as it has 100,000 ratings, 943 consumers, and 1682 movies of scale 1–5 (http://grouplens.org/datasets/movielens/). As discussed in the previous section, we presented a hybrid framework of k-means with ABC algorithm to achieve an improved movie recommendation system. Framework mentioned in the former section is used the Movielens dataset where data are considered from u1 to u5 and U_a to $U_{b.}$ To measure the performance of recommender system, mean absolute error, precision, recall, and accuracy were computed. Comprehensive analysis and behavior of recommender system framework are given below. In Table 1 and Fig. 4, precision values of our proposed system are better than those of existing methods.

$$\text{Precision} = \frac{|\text{interesting} \cap \text{Top}N|}{N} \tag{5}$$

$$\text{Recall} = \frac{|\text{interesting} \cap \text{Top}N|}{|\text{interesting}|} \tag{6}$$

We concluded from Table 2 and Fig. 5 that recall for different clusters is better for our proposed system when compared with existing methods.

Table 1 Precision for different approaches for diverse values of *k*

System/cluster	3	4	5	6	7	8	9	10
PCA-GAKM	0.4	0.39	0.38	0.379	0.37	0.36	0.359	0.355
PCA-SOM	0.38	0.39	0.38	0.36	0.35	0.348	0.33	0.326
SOM-CLUSTER	0.349	0.34	0.325	0.33	0.31	0.325	0.32	0.31
UPCC	0.31	0.3	0.295	0.29	0.285	0.28	0.27	0.25
k-means cluster	0.11	0.14	0.12	0.11	0.12	0.11	0.115	0.112
PCA-k-means	0.18	0.16	0.155	0.15	0.14	0.135	0.13	0.12
GAKM CLUSTER	0.34	0.32	0.31	0.34	0.335	0.33	0.32	0.325
ABC-KM	0.42	0.411	0.401	0.411	0.408	0.407	0.405	0.409

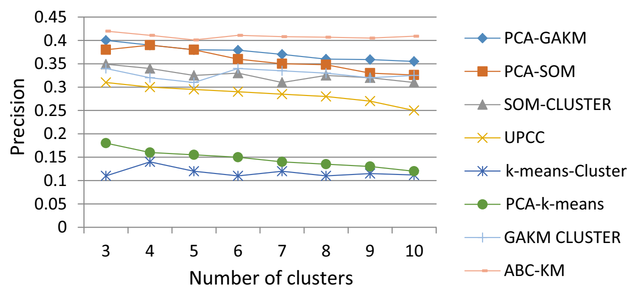

Fig. 4 Comparison of precision with various methods

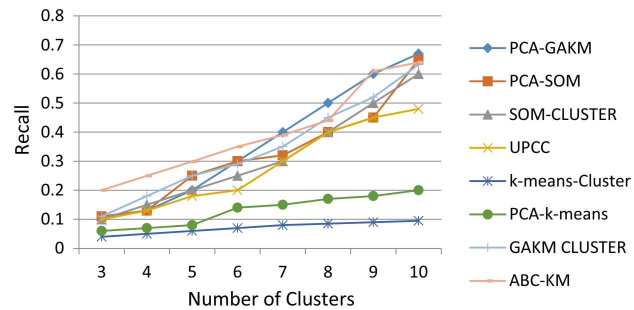

Fig. 5 Comparison of recall with various methods

We compared the performance of our proposed system with the aleady existing systems. We need to investigate the MAE of the proposed routine and then compare it against MAE of other legacy systems.

$$\mathrm{MAE} = \frac{\sum |\tilde{P}_{ij} - r_{ij}|}{M} \tag{7}$$

where M is the quantity of films, r_{ij} is the actual rating, and P_{ij} is the predicted rating of user i on movie j. We concluded that Table 3 and Fig. 6 show better MAE values for ABC-KM system when compared to other existing methods.

With an increasing amount of ratings, ABC-KM performed well in Fig. 7.

In Table 4, the performance of various methods has been presented which were performed on the Movielens dataset. We estimated the behavior of all procedures on a machine that has a configuration as an i3 processor with 1.9 GHz and 8 GB RAM. The ABC-KM system has efficient speed performance when compared to existing systems and can perform better if the proposed system is implemented in the high-configuration-processing environment.

5 Conclusion and future scope

In this article, ABC-KM with k-means and artificial bee colony optimization is proposed with the Movielens dataset that results in an improved movie recommendation system. We measured the performance of our technique with mean absolute error, precision, recall, and accuracy. The experiment outcomes on the Movielens dataset indicated that our proposed recommender system offered high performance regarding accuracy, reliability, and personalization for

Table 2 Recall for different approaches for diverse values of *k*

System/cluster	3	4	5	6	7	8	9	10
PCA-GAKM	0.11	0.13	0.2	0.3	0.4	0.5	0.6	0.67
PCA-SOM	0.11	0.13	0.25	0.3	0.32	0.4	0.45	0.65
SOM-CLUSTER	0.1	0.15	0.2	0.25	0.3	0.4	0.5	0.6
UPCC	0.1	0.13	0.18	0.2	0.3	0.4	0.45	0.48
k-means cluster	0.04	0.05	0.06	0.07	0.08	0.085	0.09	0.095
PCA-k-means	0.06	0.07	0.08	0.14	0.15	0.17	0.18	0.2
GAKM CLUSTER	0.11	0.18	0.25	0.29	0.35	0.45	0.52	0.632
ABC-KM	0.2	0.25	0.299	0.35	0.39	0.44	0.61	0.64

Table 3 MAE for different approaches for diverse standards of k

System/cluster	5	10	15	20	25	30	35	40
PCA-GAKM	0.79	0.77	0.77	0.78	0.781	0.785	0.786	0.788
PCA-SOM	0.82	0.79	0.79	0.79	0.8	0.805	0.806	0.807
SOM-CLUSTER	0.819	0.81	0.81	0.81	0.81	0.805	0.81	0.81
UPCC	0.825	0.825	0.824	0.828	0.824	0.824	0.825	0.825
k-means cluster	0.825	0.82	0.818	0.815	0.815	0.814	0.813	0.812
PCA-k-means	0.85	0.845	0.841	0.84	0.84	0.84	0.84	0.84
GAKM CLUSTER	0.815	0.805	0.804	0.804	0.804	0.804	0.803	0.803
ABC-KM	0.773	0.764	0.764	0.771	0.78	0.784	0.782	0.787

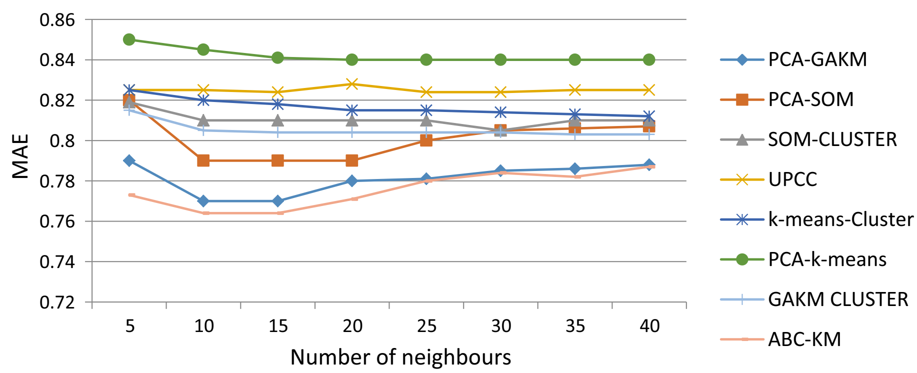

Fig. 6 Comparison of MAE with various methods

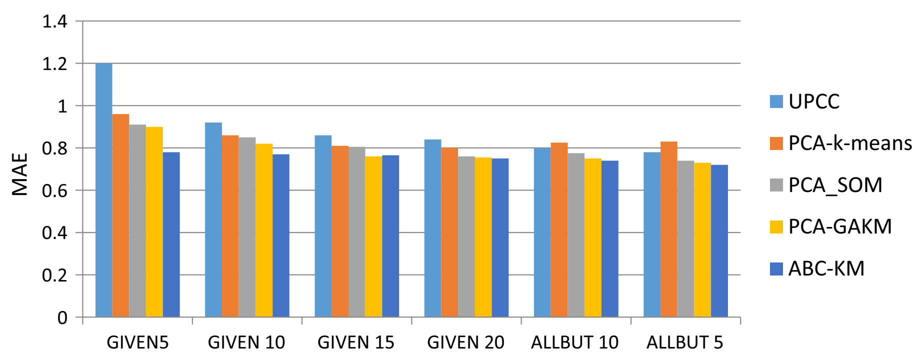

Fig. 7 MAE for the different rating level

Table 4 Comparisons of speed for different methods for Movielens dataset

Methods	PCA-GAKM	PCA-SOM	SOM-CLUSTER	UPCC	k-means cluster	PCA-k-means	GAKM CLUSTER	ABC-KM
Speed (in seconds)	26.32	141.73	76.92	159.34	16.25	55.56	355.71	53.22

movie recommendations with the specific number of clusters. For future work, our system's performance may be evaluated on advance high-configuration machine by including other important characteristics of users, such as privacy and context with cross-domain data.

Compliance with ethical standards

Conflict of interest The author declares that he has no conflict of interest.

References

1. Lu J, Wu D, Mao M, Wang W, Zhang G, Nu S (2015) Recommender system application developments. Decis Support Syst 74:12–32. https://doi.org/10.1016/j.dss.2015.03.008
2. Beel J, Gipp B, Langer S, Breitinger C (2015) Research-paper recommender systems: a literature survey. Int J Digit Libr. https://doi.org/10.1007/s00799-015-0156-0
3. Bobadilla J, Ortega F, Hernando A, Gutiérrez A (2013) Recommender systems survey. Knowl-Based Syst 46:109–132. https://doi.org/10.1016/j.knosys.2013.03.012
4. Katarya R, Verma OP (2016) Recent developments in affective recommender systems. Phys A Stat Mech Appl 461:182–190. https://doi.org/10.1016/j.physa.2016.05.046
5. Katarya R, Verma OP (2017) Efficient music recommender system using context graph and particle swarm. Multimed Tools Appl. https://doi.org/10.1007/s11042-017-4447-x
6. Colombo-Mendoza LO, Valencia-García R, Rodríguez-González A, Alor-Hernández G, Samper-Zapater JJ (2014) RecomMetz: a context-aware knowledge-based mobile recommender system for movie showtimes. Expert Syst Appl 42:1202–1222. https://doi.org/10.1016/j.eswa.2014.09.016
7. Wang Z, Yu X, Feng N, Wang Z (2014) An improved collaborative movie recommendation system using computational intelligence. J Vis Lang Comput 25:667–675. https://doi.org/10.1016/j.jvlc.2014.09.011
8. Katarya R, Verma OP (2016) A collaborative recommender system enhanced with particle swarm optimization technique. Multimed Tools Appl. https://doi.org/10.1007/s11042-016-3481-4
9. Liu H, Kong X, Bai X, Wang WEI (2015) Context-based collaborative filtering for citation recommendation. IEEE Access 3:1695–1703
10. Li Y, Zhai CX, Chen Y (2014) Exploiting rich user information for one-class collaborative filtering. Knowl Inf Syst 38:277–301. https://doi.org/10.1007/s10115-012-0583-9
11. Aguilar J, Valdiviezo-Díaz P, Riofrio G (2017) A general framework for intelligent recommender systems. Appl Comput Inform. https://doi.org/10.1016/j.aci.2016.08.002
12. Katarya R, Verma OP (2017) Effectual recommendations using artificial algae algorithm and fuzzy c-mean. Swarm Evol Comput. https://doi.org/10.1016/j.swevo.2017.04.004
13. Kunaver M, Požrl T (2017) Diversity in recommender systems—a survey. Knowl-Based Syst 123:154–162. https://doi.org/10.1016/j.knosys.2017.02.009
14. Rubio JE, Alcaraz C, Lopez J (2017) Recommender system for privacy-preserving solutions in smart metering. Pervasive Mob Comput. https://doi.org/10.1016/j.pmcj.2017.03.008
15. Katarya R, Verma OP (2016) A collaborative recommender system enhanced with particle swarm optimization technique. Multimed Tools Appl 75:1–15. https://doi.org/10.1007/s11042-016-3481-4
16. Bao J, Zheng Y, Wilkie D, Mokbel M (2015) Recommendations in location-based social networks: a survey. Geoinformatica. https://doi.org/10.1007/s10707-014-0220-8
17. Abbas A, Zhang L, Khan SU (2015) A survey on context-aware recommender systems based on computational intelligence techniques. Computing. https://doi.org/10.1007/s00607-015-0448-7
18. Katarya R, Verma OP (2016) Recommender system with grey wolf optimizer and FCM. Neural Comput Appl. https://doi.org/10.1007/s00521-016-2817-3
19. Katarya R, Verma OP (2016) An effective web page recommender system with fuzzy c-mean clustering. Multimed Tools Appl. https://doi.org/10.1007/s11042-016-4078-7
20. Liu D, Liang D, Wang C (2015) A novel three-way decision model based on incomplete information system. Knowl-Based Syst 91:32–45. https://doi.org/10.1016/j.knosys.2015.07.036
21. Mao K, Chen G, Hu Y, Zhang L (2016) Music recommendation using graph based quality model. Sig Process 120:1–8. https://doi.org/10.1016/j.sigpro.2015.03.026
22. Qiao Z, Zhang P, Cao Y, Zhou C, Guo L (2014) Improving collaborative recommendation via location-based user-item subgroup. Procedia Comput Sci 29:400–409. https://doi.org/10.1016/j.procs.2014.05.036
23. Mukherjee R, Sajja N, Sen S (2003) A movie recommendation system—an application of voting theory in user modeling. User Model User-Adapt Interact 13:5–33. https://doi.org/10.1023/A:1024022819690
24. Karaboga D, Ozturk C, Basturk B, Kıran MS, Fındık O, Karaboga D et al (2009) A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. Appl Soft Comput 11:454–462. https://doi.org/10.1016/j.asoc.2007.05.007
25. Karaboga D, Ozturk C (2011) A novel clustering approach: artificial Bee Colony (ABC) algorithm. Appl Soft Comput 11:652–657. https://doi.org/10.1016/j.asoc.2009.12.025
26. Rajasekhar A, Lynn N, Das S, Suganthan PN (2017) Computing with the collective intelligence of honey bees—a survey. Swarm Evol Comput 32:25–48. https://doi.org/10.1016/j.swevo.2016.06.001
27. Zhou A, Qu B, Li H, Zhao S, Nagaratnam P (2011) Multiobjective evolutionary algorithms: a survey of the state of the art. Swarm Evol Comput 1:32–49. https://doi.org/10.1016/j.swevo.2011.03.001
28. Mavrovouniotis M, Li C, Yang S (2017) A survey of swarm intelligence for dynamic optimization: algorithms and applications. Swarm Evol Comput 33:1–17. https://doi.org/10.1016/j.swevo.2016.12.005
29. Gotmare A, Bhattacharjee SS, Patidar R, George NV (2017) Swarm and evolutionary computing algorithms for system identification and filter design: a comprehensive review. Swarm Evol Comput 32:68–84. https://doi.org/10.1016/j.swevo.2016.06.007
30. Cosma G, Brown D, Archer M, Khan M, Pockley AG (2017) A survey on computational intelligence approaches for predictive modeling in prostate cancer. Expert Syst Appl 70:1–19. https://doi.org/10.1016/j.eswa.2016.11.006
31. Bandaru S, Ng AHC, Deb K (2017) Data mining methods for knowledge discovery in multi-objective optimization: part B—new developments and applications. Expert Syst Appl 70:119–138. https://doi.org/10.1016/j.eswa.2016.10.016
32. Rakshit P, Konar A, Das S (2017) Noisy evolutionary optimization algorithms—a comprehensive survey. Swarm Evol Comput 33:18–45. https://doi.org/10.1016/j.swevo.2016.09.002
33. Ander AR, Leser ULF, Graefe G (2017) Optimization of complex dataflows with user-defined functions. ACM Comput Surv 50:38
34. Deng S, Huang L, Xu G (2014) Social network-based service recommendation with trust enhancement. Expert Syst Appl 41:8075–8084. https://doi.org/10.1016/j.eswa.2014.07.012
35. Park DH, Kim HK, Choi IY, Kim JK (2012) A literature review and classification of recommender systems research. Expert Syst Appl 39:10059–10072
36. Bauer J, Nanopoulos A (2014) Recommender systems based on quantitative implicit customer feedback. Decis Support Syst. https://doi.org/10.1016/j.dss.2014.09.005

37. Ekstrand MD (2010) Collaborative filtering recommender systems. Found Trends® Hum Comput Interact 4:81–173. https://doi.org/10.1561/1100000009
38. Adomavicius G, Tuzhilin A (2005) Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. Adapt Web 69:253–260. https://doi.org/10.1007/978-3-540-72079-9_9
39. Sánchez-Moreno D, Gil González AB, Muñoz Vicente MD, López Batista VF, Moreno García MN (2016) A collaborative filtering method for music recommendation using playing coefficients for artists and users. Expert Syst Appl 66:234–244. https://doi.org/10.1016/j.eswa.2016.09.019
40. Wu H, Pei Y, Li B, Kang Z, Liu X, Li H (2015) Item recommendation in collaborative tagging systems via heuristic data fusion. Knowl-Based Syst 75:124–140. https://doi.org/10.1016/j.knosys.2014.11.026
41. Polatidis N, Georgiadis CK (2015) A multi-level collaborative filtering method that improves recommendations. Expert Syst Appl 48:100–110. https://doi.org/10.1016/j.eswa.2015.11.023
42. Liang X, Xia Z, Pang L, Zhang L, Zhang H (2016) Measure prediction capability of data for collaborative filtering. Knowl Inf Syst. https://doi.org/10.1007/s10115-016-0920-5
43. Ghazarian S, Nematbakhsh MA (2015) Enhancing memory-based collaborative filtering for group recommender systems. Expert Syst Appl 42:3801–3812. https://doi.org/10.1016/j.eswa.2014.11.042
44. Soares M, Viana P (2014) Tuning metadata for better movie content-based recommendation systems. Multimed Tools Appl. https://doi.org/10.1007/s11042-014-1950-1
45. Chen M-H, Teng C-H, Chang P-C (2015) Applying artificial immune systems to collaborative filtering for movie recommendation. Adv Eng Inform 29:830–839. https://doi.org/10.1016/j.aei.2015.04.005
46. Kanungo T, Mount DM, Netanyahu NS, Piatko CD, Silverman R (2002) Wu a. Y. An efficient k-means clustering algorithm: analysis and implementation. IEEE Trans Pattern Anal Mach Intell 24:881–892. https://doi.org/10.1109/TPAMI.2002.1017616
47. Ahmad A, Dey L (2007) A k-mean clustering algorithm for mixed numeric and categorical data. Data Knowl Eng 63:503–527. https://doi.org/10.1016/j.datak.2007.03.016
48. Salah A, Rogovschi N, Nadif M (2015) A dynamic collaborative filtering system via a weighted clustering approach. Neurocomputing 175:206–215. https://doi.org/10.1016/j.neucom.2015.10.050
49. Cheng L, Wang H (2014) A fuzzy recommender system based on the integration of subjective preferences and objective information. Appl Soft Comput J 18:290–301. https://doi.org/10.1016/j.asoc.2013.09.004