# Multiple-Criteria Rating Recommendation with Ordered Weighted Averaging Aggregation Operators

Hiep Xuan Huynh(✉), Loi Tan Nguyen, Hai Thanh Nguyen, and Linh Thuy Thi Nguyen

College of Information and Communication Technology, Can Tho University, Can Tho 900000, Vietnam
hxhiep@ctu.edu.vn

**Abstract.** In recent years, the Fourth Industrial Revolution in Industry 4.0 has exploded, along with the increasing development of websites, social networks, and other Internet services, leading to tremendous growth in collected data resources. Therefore, it is becoming more and more challenging to select useful information to make decisions. The recommendation systems are considered a great solution to assist humans in finding helpful information effectively and speedily. Such systems can automatically analyze, classify, select, and provide valuable information to users. Furthermore, they can explore reviews on products and services using artificial intelligence techniques to provide valuable recommendations. Users sometimes give reviews and ratings multiple times on the same products, but they differ depending on the user's mood, context, behavior, etc. Thus, the problem is accurately determining the user's rating when exploring such reviews. This work has proposed a solution for multiple-criteria rating analysis. This study has explored reviews on different criteria and integrated them into one aggregate rating by considering the similar relationship between the ratings, users, or products based on criteria in the collaborative filtering-based recommendation approach. The proposed method has performed better than traditional collaborative filtering-based methods on more than 5000 film reviews from the DePaulMovie dataset.

**Keywords:** Multiple-criteria ratings · Recommendation systems · Reviews · Context

## 1 Introduction

The problem of information overload [1] has become popular with the strong development of Internet services. The amount of information that people have access to is expanding. We can access many sources of information via email, articles, posts, advertising on social networks, e-commerce sites, etc. With the


C. V. Phan and T. D. Nguyen (Eds.): ICCASA 2022, LNICST 475, pp. 45–58, 2023.
https://doi.org/10.1007/978-3-031-28816-6_4

current expansion of information from the Internet and social networks, it will be more and more challenging to select useful information for decision-making by computer and savvy device users. A recommender system is a field of machine learning that is considered a solution to help users select information effectively and is widely applied in many fields such as science, health, education, e-commerce, entertainment, etc. The system tries to predict the "products" for the appropriate "users". In e-commerce, the recommendation system helps buyers find suitable goods, helps sellers find potential customers and boosts sales. The system recommends items according to the user's interests in entertainment and education. The recommendation system opens up the research potential of building natural systems, supporting users in decision-making.

The recommendation system is capable of automatically analyzing information, classifying, selecting, and providing users with products, goods, and services of interest through the application of statistical and artificial intelligence techniques in which machine learning algorithms play an essential role [2,3]. People divide the recommender model into many different types based on calculating the suggested results from the data. For example, the recommendation approach based on collaborative filtering is widely used in commercial fields [4–6], recommending products to users based on the similarity between users and communities. The users with the same similarity in a context can be commended for the same product. In addition, users are also suggested to use the product when most users have the same preferences on those products.

In contrast, the recommendation approach based on content filtering offers recommended products to the user when that product is similar to other products that the user in the past, which the user made as revealed in [7,8]. Furthermore, the recommendation model based on the demographic characteristics gives the recommended products to the user by using the user's demographic information such as gender, age, nationality [9,10]. The approaches based on knowledge can explore specialized knowledge, determining the product's suitability (based on descriptive attributes) with the needs or preferences of the user use in order to achieve the goal of a product that is useful to users [11–13]. The proposed integrated recommendation model aims to limit the shortcomings of the above methods [10,14,15]. Typically, integrated recommender solutions use two or more different recommendation solutions to overcome the weaknesses of each solution. Many studies prove that integrated recommender models are more accurate than single recommendation models [2]. However, these methods are also more demanding resource costs and computation time. With the diversity of recommendation models and solutions, recommendation models have been deployed and applied practically in many fields (management, commerce, health, education, and entertainment). However, technical problems still need to be further researched and perfected in the current suggested models. The recommender model based on content filtering has several disadvantages: over-specialization, feature extraction problem, and cold-Start problem; The recommendation model based on collaborative filtering suffers from limitations: new user/new product problem (cold-Start), sparsity problem, scalability problem.

problem); The demographic model based on demographic characteristics has several disadvantages: identifying user groups, determining users' preferences, and collecting personal information (demographic of users). ); The knowledge-based recommendation model has several disadvantages: the cost of knowledge acquisition, the interaction with users, and the property independence problem depending on users' preferences. Users can rate multiple times. In addition, users can give subjective preference ratings across multiple reviews of an item because it depends on the user's mood, context, behavior, etc. Thus, the problem is determining the user's rating and rating to build the most accurate and appropriate recommendation system for information/products through many reviews and ratings.

## 2 Related Work

A recommender/recommendation system (RS), as mentioned in [4,15–17], can automatically analyze, classify, predict, and provide helpful recommendations to users with information on products or services. In the recommendation system, the objects that need attention are the user (user), the item (item), and the user's feedback on the item (referred to as reviews or ratings). A recommendation system can predict how a user would rate an item, predicting the order (ranking) of items in a list from most attractive to least attractive to a user, or item (or list) item book) which is suitable for the user. A set of users (users), items (items, data items), and ratings explicitly or implicitly represent how much a user likes or dislikes an item that he or she has viewed. The recommendation system predicts the rating for an item the user has not viewed or provides a list of items the user might like. Recommendation systems are classified into the following main groups: Content-Based Recommendation Systems (CBRS), Collaborative Filtering Systems (CFS), Collaborative Filtering Systems (CFS), and Collaborative Filtering Systems (CFS). Knowledge-Based Recommendation Systems (KBRS), Hybrid Recommender System, and Context-Based Recommendation Systems (CBRS)

### 2.1 Content-Based Recommendation Systems

The content-based recommendation system [4,8,15,16] suggests to the user items that are similar to items he or she has liked in the past. This system focuses on item-specific attributes (item profile). If people make subjective judgments about some items in the past, they will also make similar judgments on other similar items in the future. So, a content-based recommendation system must learn a user's interest profile. Then, based on the user's profile, similar items that the user has liked or appreciated in the past are suggested to the user. Some techniques commonly used by content-based recommendation systems [4,8,15,17] are clustering, Bayesian classifiers, decision trees, and decision trees. Tree), rule induction, nearest neighbor method, and appropriate feedback. In content-based methods, cosine and term frequency/inverse document frequency (TF-IDF) measurements are used to measure similarity [4,15]. Some limitations of content-based recommendation systems [4,10] are: content analysis is limited because content-based techniques require explicit item descriptions; or the system recommends items whose scores are higher when compared to user profiles, so users will be suggested items that are similar to those they have previously rated, or fail to distinguish two different items represented by the same feature set; The problem is that new users - who have very few reviews - cannot get accurate recommendations.

### 2.2 Collaborative Filtering Systems

A recommendation system based on collaborative filtering [4,5,15,16] recommends a list of items for a user or predicts the rating for a particular item based on similarity measurement between users/between entries. In collaborative filtering methods, Pearson correlation coefficient, cosine-based similarity, and vector space similarity are widely used in measuring similarity between users (or items). [4,10]. Collaborative filtering is an important and popular technology for recommender systems. Collaborative filtering methods [4,15,16,18] can be grouped into two main classes: memory-based/neighborhood/heuristic (memory-based/ neighborhood-based/ heuristic-based), and model-based. The memory-based collaborative filtering approach requires that information of all ratings, items, and users be stored in the system; and directly used these assessments to make suggestions. This approach is implemented in two ways: user-based collaboration filtering and item-based collaboration filtering. The basic idea of user-based collaborative filtering is to find a set of users whose favorites are similar to a particular user (the "neighbor" of the particular user) and recommend them to others using Users or items that other users in the same group. The item-based collaborative filtering method provides a user with a recommendation for an item based on other highly correlated items ("neighbors" of the item). The techniques commonly used in memory-based collaborative filtering recommender systems [4,5,19] are the nearest neighbor, graph-based, and clustering. Meanwhile, the model-based collaborative filtering approach uses a collection of assessments to learn a model and then uses this model to make recommendations. The techniques commonly used in model-based collaborative filtering recommendation systems [4,5,20] include matrix factorization, association rule, neighbor model, neural network, and probabilistic models. Although without some of the shortcomings of content-based recommendation systems, pure collaborative filtering recommender systems have limitations [4], such as the new user problem (or the item problem). New) - the same problem as the content-based recommendation system; Sparsity problem - the number of estimates obtained is minimal compared to the number of evaluations needed for the prediction.

### 2.3 Knowledge-Based Recommendation Systems - KBRS

Knowledge-based recommendation system (KBRS) is helpful in cases where items are not frequently used. For example, in e-commerce, products are related to real estate, cars, automobiles, or tourism. In this case, the user rating matrix is not informative enough to recommend. Instead, the system can combine user ratings, product attributes, and relevant historical knowledge of similarities between user requirements to make recommendations accordingly.

## 3 Methods

### 3.1 Rating Aggregation

Rating Aggregation is calculated from the total number of ratings averaging the total number of user reviews. The aggregate function (f) represents the relationship between the overall and Multiple-criteria ratings, i.e., $r_0 = f(r_1, ..., r_k)$.

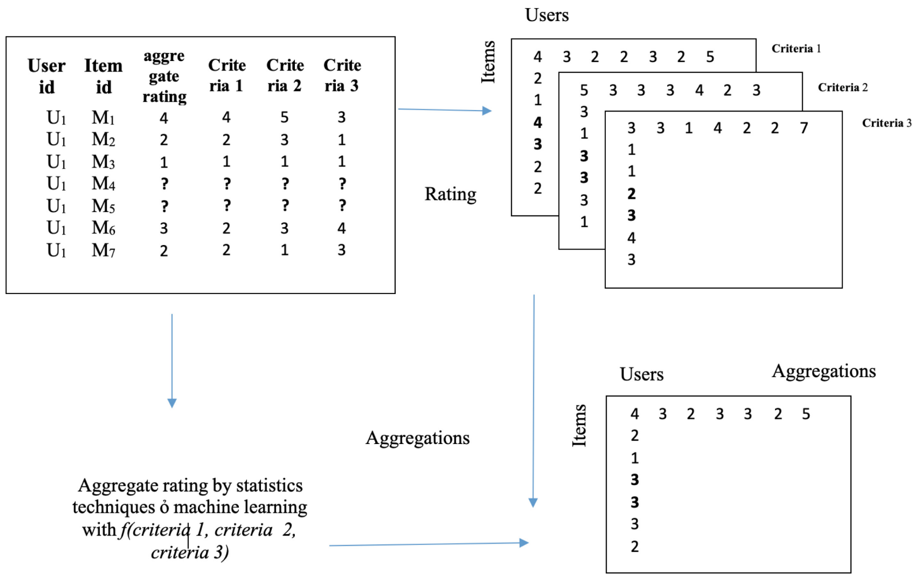


**Fig. 1.** Average aggregate rating method.

The method consists of three steps. Firstly, we estimate k individual ratings (using any recommendation technique). Then, the k-dimensional Multiple-criteria ranking problem is decomposed into k single-rank recommendation problems. In the second step, the aggregate function f is chosen by rank aggregation, statistical techniques, or machine learning techniques. Finally, the overall rating of each unevaluated data item is calculated based on the predicted k individual criterion ratings and the selected average rating aggregate as shown in Fig. 1. The approach mentioned above has applied multiple criteria ratings instead of single-criteria ratings.

### 3.2 Recommender Model with Multiple-criteria Rating

In rating evaluation, the criteria of the user or the product are the specific criteria when the user selects the product or the specific condition when the user selects the product. For example, the system suggests that users who wish for a tour booking support system advise choosing a deserved vacation. In that case, the time factor and the person accompanying them will significantly affect the user when determining the destination for the trip. Alternatively, a movie that many users choose to watch in a particular space and time. From the above description, we can see that the information about the user or product criteria depends on each specific consulting problem. However, to model the recommendation problem based on the criteria, the information about the user's or product's criteria is defined as follows: We consider the set $U = \{u_1, u_2, ..., u_n\}$ including of $n$ users while the set of $I = \{i_1, i_2, ..., i_m\}$ being $m$ products; $C = \{c1, c2, ..., ck\}$ is the set of $k$ criteria attributes when the user selects the products. When user $u_a$ selects product $i_j$, or when product $i_j$, is selected by user $u_a$, the values of the criteria attributes are specified as follows (as shown in Eq. 1):

$$C_{a,j} = c1_{a,j}, c2a, j, ..., cka, j \tag{1}$$

From the criteria, we construct a criterion similarity matrix between users or products that is a symmetric matrix with structure, i.e., rows and columns of the matrix are users or products. In addition, cells of the matrix (row and column intersection) are the criterion similarity value between two users or two products on row and column, respectively. Given the user set $U = u_1, u_2, ..., u_n$ and the product set $I = i_1, i_2, ..., i_m$. Then, the criteria similarity matrix between users is determined as Eq. 2.

$$matrix_{sim}(C_u) = \begin{pmatrix} 1 & su_{1,2} & ... & su_{1,n} \\ su_{2,1} & 1 & ... & su_{2,n} \\ ... \\ su_{n,1} & su_{n,2} & ... & 1 \end{pmatrix} \tag{2}$$

where $su_{a,b}$ is the value of similarity criteria between two users $u_a$ and $u_b$. The criteria similarity matrix between products is determined as Eq. 3.

$$matrix_{sim}(C_u) = \begin{pmatrix} 1 & si_{1,2} & ... & si_{1,m} \\ si_{2,1} & 1 & ... & si_{2,m} \\ ... \\ si_{m,1} & si_{m,2} & ... & 1 \end{pmatrix} \tag{3}$$

where $si_{j,h}$ is the value of similarity criteria between two products $i_j$ and $i_h$

### 3.3 Recommendations Based on Similarity Between Users/Items

The similarity-based recommendation model between users is defined as follows.

Let $U = \{u_1, u_2, ..., u_n\}$ be the set of n users; $I = \{i_1, i_2, ..., i_m\}$ is the set of m products; $C = \{c1, c2, ..., ck\}$ denotes the set of k criteria attributes when the user selects the products; $R = r_{c,j}$ exhibits user U's rating matrix for items I in criterion C, with each row representing one user $u_c, (1 \leq c \leq n)$, each column representing for an item $i_j, (1 \leq j \leq m)$, $r(c, j)$ is the rating value of user $u_c$ for product $i_j$ in the criterion $C_{cj}$. N is the number of products with high similarity and $u_a \in U$ is the user who needs to suggest with criteria: $C_{a,j} = \{c1_{a,j}, c2_{a,j}, ..., ck_{a,j}\}$.

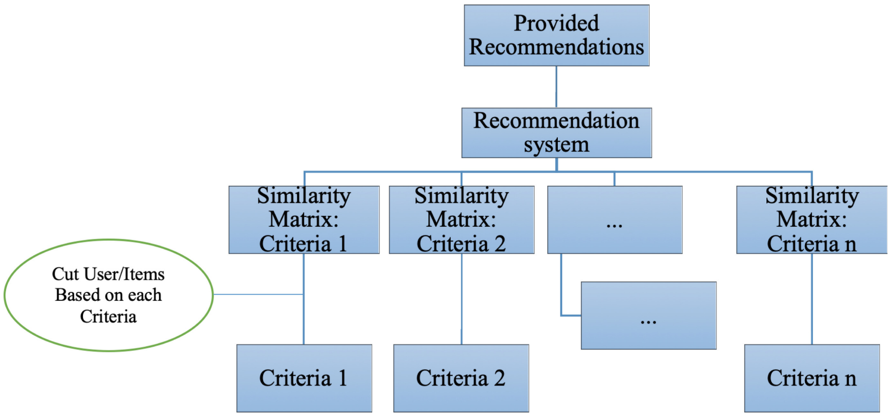


**Fig. 2.** Recommendation system based on the similarity between users/items.

The recommendation systems are based on the similarity of user criteria as presented in Fig. 2. In the first stage, the criteria are conducted from a user-based ranking matrix using the User splitting technique. Then, based on the criteria attributes, we build the criteria similarity matrix between users; Finally, the user-criteria similarity-based recommendation model is built based on the integration matrix between the User-Based Rating Matrix and the user-criteria similarity matrix. Based on input which includes Rating from users/items;criteria;users/items for recommendations: $u_\alpha$. To provide N products recommended for $u_\alpha$, the algorithm based on the similarity between criteria according to the user (in the case of items we present in *brackets*) is exhibited as follows:

- Step 1: Build user-based similarity matrix- $SU_R$ (items-based similarity matrix $SI_R$ in the case we based on items):
  - Handle cutting users (items);
  - Build users-based similarity matrix (items-based similarity matrix);
- Step 2: Build a similarity matrix of criteria according to users: $SU_C$ (similarity matrix of criteria according to items: $SI_C$):
  - Similarity of criteria between 2 users (2 items);
  - Building similarity matrix based on user criteria (item criteria);
- Step 3: Build the integrated similarity matrix: $SU_I = SU_R + SU_C$ (if we use "item", it will be $SI_I = SI_R + SI_C$);

– Step 4: Build User-Based Collaborative Filtering based on the average integrated similarity matrix of the Criteria: UBCFC (in the case with items, i.e., Item-Based Collaborative Filtering based on the average integrated similarity matrix of the Criteria (IBCFC));
– Step 5: Define a list of similar products for user $u_\alpha$;
– Step 6: Present the user $u_a N$ with the item with the highest similarity value;

### 3.4 The Recommendation Approach Integrated with OWA, WOWA

The study evaluates the rating of the criteria of the user or the product, which is the specific criterion when the user selects the product or the specific condition when the product is selected by the user integrating Ordered Weighted Average (OWA) [21,22], and Weighted OWA (WOWA) [23] into the process proposed to estimate user preferences, predicting into a "multi-dimensional" rating as exhibited in Fig. 3.

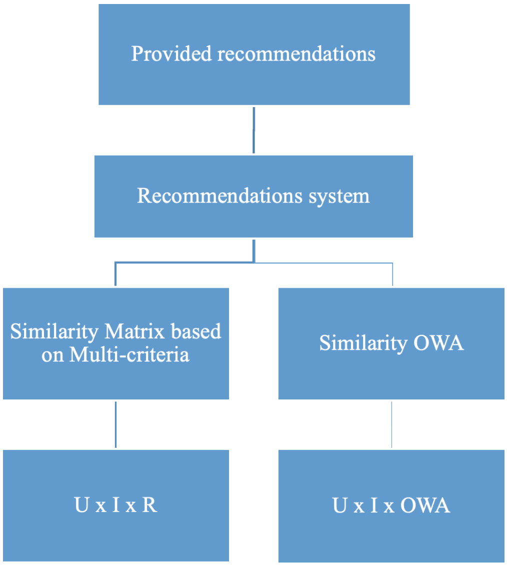


**Fig. 3.** Recommendation system integrated with OWA.

## 4 Experiments

### 4.1 Data Description

We evaluated the proposed method on DePaulMovie dataset. DePaulMovie[1] was updated in 2018 by the Pepoul Center for Data Science, including a survey of

[1] https://github.com/JDonini/depaulmovie-recommender-system.

students ranking movies by different time, place, and companions. The dataset consists of 5043 ratings from 97 students of 319 movies based on three context attributes Time, Location, and Companion (each context is a criterion). The rank attribute has a positive integer value ranging from 1 (very unsatisfied) to 5 (very satisfied), in which the value 1 appears in 829 reviews while the value of 2 is in 625 reviews, etc.

### 4.2 Contextual Data Processing on DePaulMovie Dataset

From students' ratings on movies, we construct a real numeric rating data matrix for the DePaulMovie dataset with a structure of 97 rows (corresponding to 97 users) and 319 columns (corresponding to 319 movies/items) with 5043 rating values. The criteria attributes are used to cut users according to each specific criteria when the user chooses to watch movies to build a similarity matrix of criteria according to users.

There are three considered contexts Time, Location, and Companion (each context is considered as a criterion). Each criterion contributes to a user matrix with 97 rows and 319 columns so that the data has three matrices of the user. In addition, the criteria attribute crop products according to specific criteria when the user selects the movies to watch to build a product-specific criterion similarity matrix.

### 4.3 Evaluation of Multiple-Criteria-Based Approach

The dataset was divided into 5-fold-cross-validation to evaluate the effectiveness of the recommendation system. The method with a Multiple-criteria ranking was conducted from the traditional collaborative filtering model. For example, if the data is divided by 80% for the training set, and the test set occupies 20%, the results of the training data set can contain 3723 evaluations from 71 users (71 rows $\times$ 319 columns). The test set includes 26 users (26 rows $\times$ 319 columns) with 1320 reviews. The experiments are done with the support from some libraries in R, such as Recommenderlab [24,25], agop [26], and the performance is measured by Mean Square Error (RMSE), Mean Square Error (MSE), Mean Absolute Error (MAE) on the test set.

**Table 1.** Comparison between UBCFC and UBCF in various metrics.

| Method | RMSE | MSE | MAE |
|---|---|---|---|
| UBCFC | 1.409295 | 1.986112 | 1.114970 |
| UBCF | 1.410167 | 1.988571 | 1.114970 |

Table 1 presents the results and compares UBCFC and the traditional User-Based Collaborative Filtering (UBCF) method in RMSE, MSE, and MAE on the test set. This result shows that the approach that integrated the user-criteria similarity matrix for providing recommendations, UBCFC, has lower error rates than UBCF.

**Table 2.** Comparison between IBCFC and IBCF in various metrics.

| Mô hình | RMSE | MSE | MAE |
|---|---|---|---|
| IBCFC | 1.592475 | 2.535977 | 1.220261 |
| IBCF | 1.598317 | 2.543797 | 1.230216 |

The recommendation model is based on the similarity of user-selected product criteria (IBCFC) and the traditional Product-Based Collaborative Filtering Recommendation Model (IBCF). From the experimental results, measure error parameters: Square root of mean square error (RMSE), mean square error (MSE), mean absolute error (MAE) in Table 2. This result shows that the model that integrates the user-selected product criteria similarity matrix (IBCFC) has lower error parameters than the traditional Product-Based Collaborative Filter Recommendation Model (IBCF).

**Table 3.** Comparison between IBCF_OWA and IBCF in RMSE, MSE and MAE.

| Methods | RMSE | MSE | MAE |
|---|---|---|---|
| IBCF_OWA | 1.431843 | 2.050174 | 1.179211 |
| IBCF | 1.598317 | 2.543797 | 1.230216 |

The OWA Integrated Recommendation method (IBCF_OWA) and the traditional Item-Based Collaborative Filtering Recommendation Model (IBCF) with results in Square root of mean square error (RMSE), mean square error (MSE), mean absolute error (MAE) are revealed in Table 3. In addition, they show that the OWA integration model (IBCF_OWA) has lower error parameters than the traditional Product-Based Collaborative Filtering Recommendation Model (IBCF) running on the same DePaulMovie dataset.

To evaluate the effectiveness of the methods, we compare the proposed method with the traditional collaborative filtering model UBCF and IBCF in the ROC metric based on the Precision/Recall ratio on the DePaulMovie dataset.

The recommendation strategy is based on the similarity of user criteria for products (UBCFC) and the traditional User-Based Collaborative Filtering Recommendation Model (UBCF). The results of comparing the model's accuracy are shown in (Fig. 4) with the number of films introduced to users of the models increasing gradually from 1 to 10. The results show that the Precision/Recall

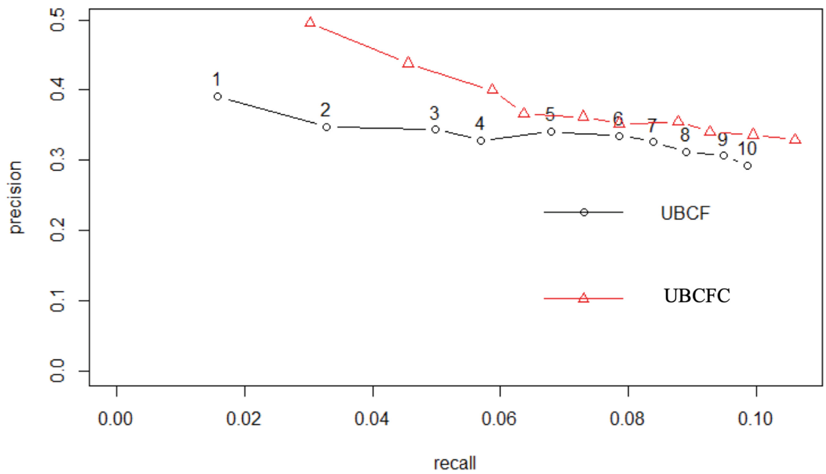


**Fig. 4.** The comparison results on DePaulMovie dataset between UBCFC and UBCF.

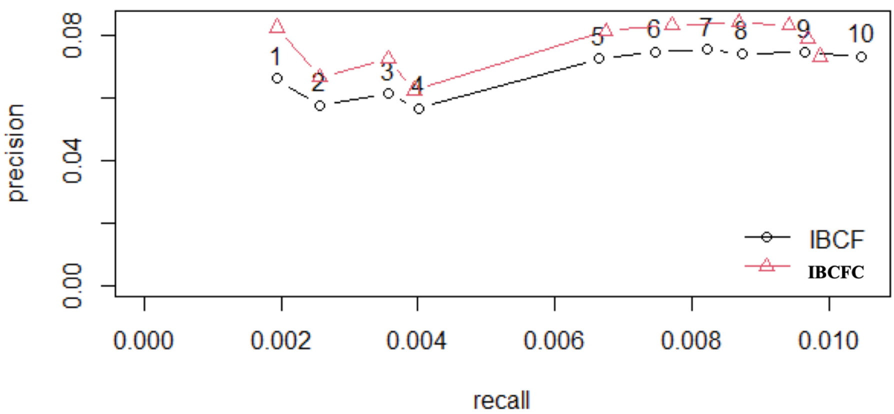


**Fig. 5.** The comparison results on DePaulMovie dataset between IBCFC and IBCF.

ratio of the models increased gradually from 1 to 10. the UBCFC model is relatively higher than the Precision/Recall ratio of the two UBCF models. Therefore, the recommendation approach based on the similarity of user criteria for products (UBCFC) can improve the accuracy of the recommender system.

The proposed method compares user-selected product criteria (IBCFC) and the traditional Product-Based Collaborative Filtering Recommendation Model (IBCF). The results are presented in Fig. 5 with the number of films introduced to users of the models that increase gradually from 1 to 10. The results show that the Precision/Recall ratio of the models is increased. Furthermore, the IBCFC model is relatively higher than the Precision/Recall ratio of the two IBCF models. As observed, the recommendation methods based on the similarity of selected product criteria (IBCFC) can improve the accuracy of the recommender system.

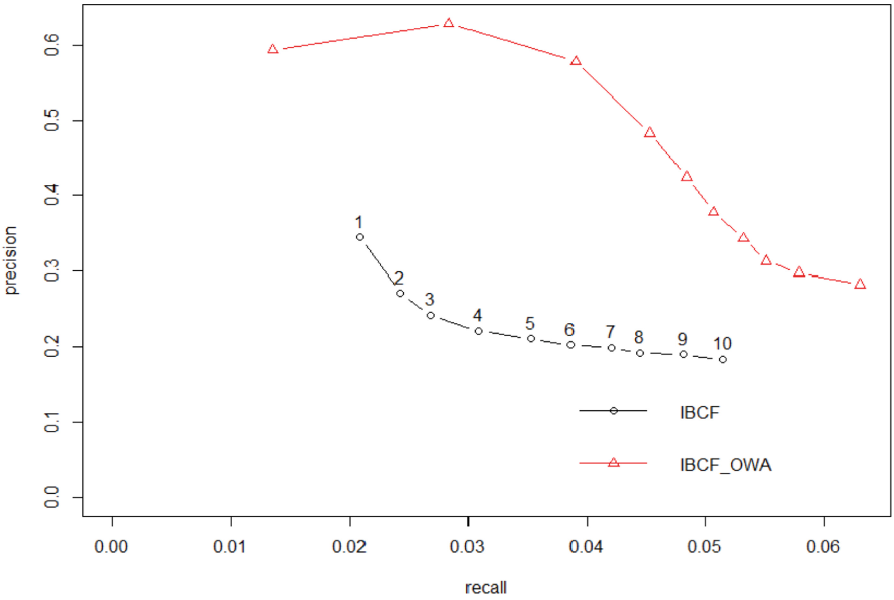


**Fig. 6.** The comparison results on DePaulMovie dataset between IBCF and IBCF_OWA.

The OWA integrated Item-Based Collaborative Filtering Recommendation Model (IBCF_OWA) and the traditional Item-Based Collaborative Filtering Recommendation Model (IBCF) are compared in Fig. 6. Comparing the accuracy of the model reveals that the number of films introduced to users of the models increased gradually from 1 to 10. The results show that the Precision/Recall ratio of the models is increased. The OWA model (IBCF_OWA) is relatively higher than the Precision/Recall ratio of the two IBCF models. It shows that the OWA integrated Item-Based Collaborative Filtering Recommendation Model can improve the accuracy of the recommender system.

## 5 Conclusion

Experimental results are evaluated on film reviews of the DePaulMovie dataset with a recommendation system's desired results. The study proposes a method to improve the accuracy of the collaborative filtering recommender model by considering the similar relationship between users or products based on the criteria in building the model. Collaborative filtering consulting. The recommendations are calculated based on the integration of similarity values, i.e., similarity based on user's criteria for products and similarity based on product criteria selected by users. The Selected and matched items/users are based on OWA integration. Experimentation on the DePaulMovie dataset shows that the proposed model has higher accuracy than the traditional collaborative filtering recommender model. From these experimental results, it can be confirmed that the recommendation model based on Multiple-criteria ratings is applicable in practice.

## References


1. Toffler, A.: Future Shock. Bantam Books (1970)
2. Nghia, P.Q., Phuong, D.H., Hiep, H.X.: Lua chon mo hinh va tham so cho bai toan tu van loc cong tac dua tren do thi. Can Tho University, Journal of Science, p. 171 (2017). https://doi.org/10.22144/ctu.jsi.2017.023
3. Bobadilla, J., Ortega, F., Hernando, A., GutiéRrez, A.: Recommender systems survey. Know. Based Syst. **46**, 109–132 (2013). https://doi.org/10.1016/j.knosys.2013.03.012
4. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Trans. Knowl. Data Eng. **17**(6), 734–749 (2005)
5. Schafer, J.B., Frankowski, D., Herlocker, J., Sen, S.: Collaborative filtering recommender systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web. LNCS, vol. 4321, pp. 291–324. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72079-9_9
6. Ekstrand, M.D., Riedl, J.T., Konstan, J.A.: Collaborative filtering recommender systems. Found. Trends Hum. Comput. Interact. 4(2), 81–173 (2011). https://doi.org/10.1561/1100000009
7. Cantador, I., Bellogín, A., Vallet, D.: Content-based recommendation in social tagging systems. In: Proceedings of the Fourth ACM Conference on Recommender Systems, ser. RecSys 2010, pp. 237–240. New York, NY, USA: Association for Computing Machinery (2010). https://doi.org/10.1145/1864708.1864756
8. Pazzani, M.J., Billsus, D.: Content-based recommendation systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web. LNCS, vol. 4321, pp. 325–341. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72079-9_10
9. Krulwich, B.: Lifestyle finder-intelligent user profiling using large-scale demographic. AI Mag. 10 (1997)
10. Cobos, C., et al.: A hybrid system of pedagogical pattern recommendations based on singular value decomposition and variable data attributes. Inf. Process. Manage. **49**(3), 607–625 (2013). https://doi.org/10.1016/j.ipm.2012.12.002
11. Felfernig, A., Teppan, E., Gula, B.: Knowledge-based Recommender technologies for marketing and sales. Int. J. Pattern Recogn. Artif. Intell. **21**(02), 333–354 (2007). https://doi.org/10.1142/s0218001407005417
12. Shambour, Q., Lu, J.: A trust-semantic fusion-based recommendation approach for e-business applications. Decis. Support Syst. **54**(1), 768–780 (2012). https://doi.org/10.1016/j.dss.2012.09.005
13. Nguyen, T.T.S., Lu, H.Y., Lu, J.: Web-page recommendation based on web usage and domain knowledge. IEEE Trans. Knowl. Data Eng. **26**(10), 2574–2587 (2014)
14. Lucas, J.P., Luz, N., Moreno, M.N., Anacleto, R., Figueiredo, A.A., Martins, C.: A hybrid recommendation approach for a tourism system. Expert Syst. Appl. **40**(9), 3532–3550 (2013). https://doi.org/10.1016/j.eswa.2012.12.061
15. Burke, R.: Hybrid web recommender systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web. LNCS, vol. 4321, pp. 377–408. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72079-9_12

16. Felfernig, A., Jeran, M., Ninaus, G., Reinfrank, F., Reiterer, S., Stettinger, M.: Basic approaches in recommendation systems. In: Robillard, M.P., Maalej, W., Walker, R.J., Zimmermann, T. (eds.) Recommendation Systems in Software Engineering. LNCS, pp. 15–37. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-45135-5_2
17. Leskovec, J., Rajaraman, A., Ullman, J.D.: Mining of Massive Datasets. Cambridge University Press, Cambridge (2014)
18. Breese, J.S., Heckerman, D., Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, ser. UAI 1998, pp. 43–52 San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. (1998)
19. Desrosiers, C., Karypis, G.: A comprehensive survey of neighborhood-based recommendation methods. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook. LNCS, pp. 107–144. Springer, Boston, MA (2011). https://doi.org/10.1007/978-0-387-85820-3_4
20. Koren, Y., Bell, R.: Advances in collaborative filtering. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook. LNCS, pp. 145–186. Springer, Boston, MA (2011). https://doi.org/10.1007/978-0-387-85820-3_5
21. Yager, R.R., Kacprzyk, J.: The Ordered Weighted Averaging Operators. Springer, Cham (1997)
22. Csiszar, O.: Ordered weighted averaging operators: a short review. IEEE Systems, Man Cybern. Mag. **7**(2), 4–12 (2021)
23. Torra, V.: The WOWA operator: a review. In: Yager, R.R., Kacprzyk, J., Beliakov, G. (eds.) Recent Developments in the Ordered Weighted Averaging Operators: Theory and Practice. Studies in Fuzziness and Soft Computing, vol. 265, pp. 17–28. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-17910-5_2
24. Gorakala, S.K., Usuelli, M.: Building a Recommendation System with R. Packt Publishing, Birmingham (2015)
25. Hahsler, M. : recommenderlab: a framework for developing and testing recommendation algorithms (2011)
26. Gagolewski, M., Cena, A.: AGOP: aggregation operators package for R (2014). http://agop.rexamine.com/