Multi-criteria Collaborative Filtering Model Based on Contextual Rating Data

Hiep Xuan Huynh [1], Nghia Quoc Phan [2,*], and Loi Tan Nguyen [3]

[1] College of Information and Communication Technology, Can Tho University, Can Tho, Vietnam; Email: hxhiep@ctu.edu.vn (H.X.H.)
[2] Faculty of Engineering and Technology, Tra Vinh University, Tra Vinh, Vietnam
[3] People's Committee of Phu Tan District, Camau, Vietnam; Email: tanloi1978@gmail.com (L.T.N.)
[*]Correspondence: nghiatvnt@tvu.edu.vn (N.Q.P.)

***Abstract*—Numerous research studies have emphasized the significance of contextual information when it comes to recommender models. This importance is especially evident in the realm of e-commerce platforms, where recommender systems have been effectively suggesting products and services to users by integrating contextual data into their models. By doing so, these systems can better understand user preferences and behaviors during transactions on the platform. As a result, a growing number of platforms are now collecting evaluation values for products and services based on various user contexts, leading to the emergence of multi-context-based rating datasets. This presents a valuable opportunity to implement multi-criteria collaborative filtering models, which we propose as a solution. Our approach involves integrating user contextual rating data and conducting experiments using two sets of contextual evaluation datasets: De Paul Movie and In Car Music. The results demonstrate that the multi-criteria collaborative filtering model outperforms the single-context-based collaborative filtering model in terms of accuracy. This study opens up promising avenues for future research aimed at further enhancing recommendation accuracy for customers on online sales platforms.**

***Keywords*—rating matrix, contextual rating data, multi-criteria model**

I. Introduction

The commercial industry has experienced a significant boost due to advancements in information and communication technology. Nowadays, numerous trading companies are tapping into the potential of online sales and platforms to reach their target customers. However, this has led to an overwhelming amount of information on product pages, causing confusion among buyers. To enhance customer experience and offer better services, many online trading sites have integrated recommender models. These models assist customers in making informed decisions, saving them valuable time when using online services. Presently, user rating data is widely employed in suggesting personalized recommendations for online sales and services [1]. Nevertheless, the current recommender models primarily rely on mining rating data using collaborative filtering models [2]. These models are categorized as single-criteria recommenders, as they mainly depend on overall product rating values and overlook individual user preferences in specific contexts. Consequently, the contextuality of each user is disregarded, particularly when it comes to evaluating products.

In recent times, there has been a shift towards using contextual ratings rather than just overall ratings in online reviews [3]. Additionally, an increasing number of commercial websites now allow users to rate products based on specific contexts [3]. As a result, there is a growing trend in developing recommender models that leverage contextual rating data to enhance the accuracy of collaborative filtering models. The adoption of multi-criteria collaborative filtering models has emerged as a more effective approach, considering users' preferences across various contexts [4]. By doing so, this model better comprehends users' preferences for product purchases or service usage in specific scenarios. In essence, integrating information from each user's contextual rating data enables a closer alignment with their individual preferences [5].

The objective of this study is to construct a multi-criteria collaborative filtering model that incorporates contextual rating data to enhance the performance of existing collaborative filtering models based solely on overall rating data. This involved reimagining the collaborative filtering models to incorporate users' contextual rating data. Through a comparative analysis, the experimental results demonstrated that the multi-criteria collaborative filtering models outperformed the single-criteria collaborative filtering models in terms of performance.

This article is structured into seven distinct sections. Section I offers an overview of the research problem at hand. Section II provides a concise introduction to collaborative filtering models, context-based recommender models, and multi-criteria recommender models. Section III delves into the presentation of the contextual user rating matrix. Section IV elaborates on multi-criteria collaborative filtering models designed for contextual rating data. Section V introduces various

Manuscript received February 12, 2023; revised March 30, 2023; accepted August 4, 2023; published December 7, 2023.

methods employed for the assessment of collaborative filtering models. In Section VI, we present the experimental outcomes of the proposed models, utilizing the DePaulMovie and InCarMusic datasets. Finally, Section VII encapsulates a summary of the achieved results.

II. Related Work

Numerous research studies have delved into recommender models that incorporate rating matrices and contextual data related to users' product purchases or service usage. These models include collaborative filtering recommender models [6], context-aware recommender models [7], context-similarity collaborative filtering recommender models, and multi-criteria collaborative filtering recommender models [8].

The collaborative filtering model has found significant success in various e-commerce websites. It was first introduced in the article "Group Lens: an open architecture for collaborative filtering of Netnews" in 1994 [9] and was adopted by Amazon.com to introduce users to their products in 2003 [10]. This model suggests new products or services to users by predicting their rating values based on a rating matrix, which represents user ratings for different products or services. This prediction is achieved using machine learning algorithms [11]. However, the collaborative filtering model has shown limitations in the following scenarios: for new users with no ratings yet, for new products without any user ratings, and in cases where data is sparse with too few user ratings for certain products or services.

The context-based collaborative filtering model represents the advancement of the traditional collaborative filtering model. Like its predecessor, this model also relies on user rating data to suggest product or service categories. However, the context-based collaborative filtering model incorporates additional contextual factors in predicting user rating values for specific products or services. For instance, when recommending movies to users, genre becomes an essential contextual factor, while time and geographical distance play crucial roles in tourist attraction recommendations. The context-based collaborative filtering models are categorized into three groups based on how they incorporate context information in the prediction process. Firstly, the Contextual pre-filtering group employs the user's contextual information to select relevant data sets before making predictions. Secondly, the contextual post-filtering group tailors the prediction results for each user by using contextual information. Lastly, the Contextual modeling group directly integrates contextual information into the process of predicting user rating values.

The context-similarity collaborative filtering recommender model presents an alternative approach to the context-based recommender model. This model recommends products or services to users by integrating two distinct values. The first value is derived from user rating data for the products or services, while the second value is calculated based on context attributes. The key distinction between the context-similarity collaborative filtering model and the context-based collaborative filtering model lies in how they utilize context attributes. In the context-similarity collaborative filtering model, context attributes are employed to determine the similarity between two users or two products, leading to the construction of a context-similarity matrix. On the other hand, the context-based collaborative filtering model uses context attributes to filter data or adjust model outcomes. There are two variations of the context-similarity collaborative filtering model: the Context-similarity User-Based Collaborative Filtering (CUBCF) model, where similarity values are calculated based on users, and the Context-similarity Item-Based Collaborative Filtering (CIBCF) model, where similarity values are calculated based on products or services.

The development of the multi-criteria collaborative filtering model revolves around the collection of users' rating data. In the present day, numerous e-commerce systems are gathering multi-attribute (multi-context) rating data, which offers more comprehensive insights into a user's preferences for products in various contexts, surpassing the limited scope of an overall rating value that represents their general opinion of all products. By leveraging multi-context rating data, the multi-criteria collaborative filtering model achieves a more profound understanding of user preferences when it comes to product purchases or service usage in specific contexts. This marks a growing trend toward maximizing user personalization in recommender models. The focus on multi-context data has become instrumental in enhancing the accuracy and effectiveness of recommender systems, leading to better user experiences and improved recommendations.

III. Contextual User Rating Matrix

A contextual user rating matrix serves as a repository for a user's ratings of products they have purchased or services they have used in specific contexts. For instance, when a user rates a movie, their rating can be significantly influenced by factors such as the time and location of the viewing. As described above, the composition of the contextual user rating matrix varies according to each distinct recommendation problem. The key differentiation between the overall rating matrix and the contextual user rating matrix is exemplified in the following manner [12]:

$$Users \times Items \rightarrow R_0 \times R_1 \times R_2 \ldots \times R_k \tag{1}$$

where R_0 is the overall rating matrix and R_i is the contextual user rating matrix for context i ($i = 1, \ldots, k$).

IV. Multi-criteria Collaborative Filtering Model for Contextual Rating Data

In this section, two multi-criteria collaborative filtering models are proposed for contextual rating data. The first is the User-Based Multi-Criteria Collaborative Filtering (MC-UBCF) recommender model and the second is the Item-Based Multi-Criteria Collaborative Filtering (MC-

IBCF) recommender model. In particular, the former is designed according to the structure of the traditional single-criteria UBCF model while the latter is developed according to the structure of the single-criteria traditional IBCF model.

A. MC-UBCF Model

The MC-UBCF model is designed with the following structure:

Given the following sets: $U = \{u_1, u_2, \dots, u_n\}$ is the set of users; $I = \{i_1, i_2, \dots, i_m\}$ is the set of items; $C = \{c_1, c_2, \dots, c_k\}$ is the set of attributes contextual when the user selects items; $R = \{R_1, R_2, \dots, R_k\}$ are contextual user rating matrices.

The MC-UBCF model is represented as follows:

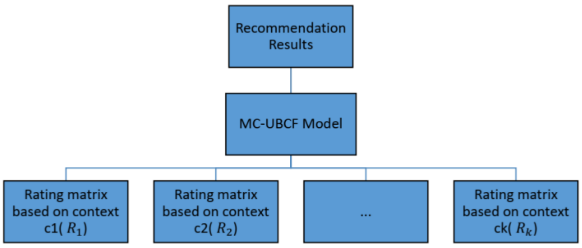

Figure 1. The MC-UBCF model.

Fig. 1 presents the MC-UBCF model, in which the rating matrices R_1, R_2, …, and R_k are built from contextual user rating data. The recommender model is designed based on contextual rating matrices. This model considers each contextual rating matrix as the MC-UBCF model. These criteria are included in the collaborative filtering model by using the integration method to form the integration matrix, from which the recommendation results of the model will be predicted based on the integration matrix according to the user similarity.

Algorithm 1. MC-UBCF recommender

Input: Contextual user rating matrices, Current user u_a;
Output: N items used to introduce to user u_a;
Begin

Step 1: Build an integrated rating matrix

$$R_I = w_1 \times R_1 + w_2 \times R_1 R_2 + \cdots + w_k \times R_1 R_k$$

(The $\mathrm{w_i}$parameter has a value between 0 and 1 depending on the importance of the context the $\mathrm{C_i}$, such that $\sum_{\mathrm{i}=1}^{\mathrm{k}} \mathrm{w_i} = 1$. This weight is determined based on contextual user rating data.)

Step 2: Build a user similarity matrix from the m integration matrix

$$Matrix_{sim}(R_I) = \begin{pmatrix} su_{1,1} & su_{1,2} & \cdots & su_{1,n} \\ su_{2,1} & su_{2,2} & \dots & su_{2,n} \\ \vdots & . & \ddots & \vdots \\ su_{n,1} & su_{n,2} & \cdots & su_{n,n} \end{pmatrix}$$

$\mathrm{su_{i,j}}$ is a similarity value between the user $\mathrm{u_i}$ and user $\mathrm{u_j}$. This value is calculated by similarity measures such as “jaccard”, “dice”, “cosine”, “Euclid”, “pearson”. The choice of the measures will depend on the particular rating dataset.

Step 3: Build MC-UBCF model based on user similarity matrix;

Step 4: Identify a list of similar items for the user u_a;

Step 5: Recommend user u_a N items the highest similarity value;

End;

B. MC-IBCF Model

The MC-IBCF model is designed with the following structure:

Given the following sets: $U = \{u_1, u_2, \dots, u_n\}$ is the set of users; $I = \{i_1, i_2, \dots, i_m\}$ is the set of items; $C = \{c_1, c_2, \dots, c_k\}$ is the set of attributes contextual when the user selects items; $R = \{R_1, R_2, \dots, R_k\}$ are contextual user rating matrices.

The MC-IBCF model is shown as follows:

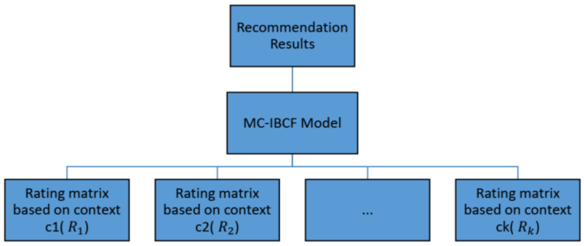

Figure 2. The MC-IBCF model.

Fig. 2 presents the MC-IBCF model, in which the rating matrices R_1, R_2, …, and, R_k are built from contextual user rating data. The recommender model is designed based on contextual rating matrices. This model considers each contextual rating matrix as one criterion in the MC-IBCF model. These criteria are included in the collaborative filtering model through the integration method to form the integration matrix, from which the recommendation results of the model will be predicted based on the integration matrix according to the item similarity.

Algorithm 2. MC-IBCF recommender

Input: Contextual user rating matrices, Current user u_a;
Output: N items used to introduce to user u_a;
Begin

Step 1: Build an integrated rating matrix

$$\mathrm{R_I} = \mathrm{w_1} \times R_1 \mathrm{R_1} + \mathrm{w_2} \times R_1 \mathrm{R_2} + \cdots + \mathrm{w_k} \times R_1 \mathrm{R_k}$$

The $\mathrm{w_i}$parameter has a value between 0 and 1 depending on the importance of the context $\mathrm{C_i}$, such that $\sum_{\mathrm{i}=1}^{\mathrm{k}} \mathrm{w_i} = 1$. This weight is determined based on contextual user rating data.

Step 2: Build an item similarity matrix from the integration matrix

$$Matrix_{sim}(R_I) = \begin{pmatrix} si_{1,1} & si_{1,2} & \cdots & si_{1,m} \\ si_{2,1} & si_{2,2} & \dots & si_{2,m} \\ \vdots & . & \ddots & \vdots \\ si_{m,1} & si_{m,2} & \cdots & si_{m,m} \end{pmatrix}$$

$\mathrm{si_{a,b}}$ is a similarity value between the item $\mathrm{i_a}$ and item $\mathrm{i_b}$. This value is calculated by similarity measures such as “jaccard”, “dice”, “cosine”, “Euclid”, “pearson”. The choice of the measures will depend on the particular rating dataset.)

Step 3: Build MC-IBCF model based on item similarity matrix;

Step 4: Identify a list of similar items for the user $\mathrm{u_a}$;

Step 6: Recommend user u_aN items the highest similarity value;

End;

V. Evaluating Prediction Accuracy for Collaborative Filtering Models

Two commonly used methods are employed to assess the performance of collaborative filtering models. The first method involves measuring the accuracy of the model's rating predictions, utilizing metrics such as Mean Squared Error (MSE), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE) [13]. The second method revolves around evaluating the model's prediction outcomes, employing metrics like Accuracy, Recall, F-score, and ROC curve [13]. Both evaluation approaches operate under the assumption that a well-performing model on existing user data will make accurate predictions for new users. To accomplish this, the experimental datasets are typically divided into two sets: one for training the model and the other for testing its performance. A successful model yields similar predicted rating values to the actual ratings of users in the test set or provides predictions of products that users have purchased with high rating values in the test set [13].

A. *Evaluation is Based on Measuring the Accuracy of Rating Predictions*

This approach involves computing the disparity between the model's predicted rating values and the actual rating values given by users, making use of numerical range for scoring models. Error measurements in statistics, such as Mean Square Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Normalized Mean Absolute Error (NMAE), are commonly used in this method. Both MSE and RMSE penalize larger prediction errors more significantly compared to MAE. However, RMSE is more widely utilized than MSE when assessing the performance of random models because it shares the same units as the dependent variable. Additionally, being a differentiable function, MSE facilitates mathematical operations, making it preferable over non-differentiable functions like MAE in many models. Due to its ease of interpretation and calculation, RMSE is frequently used as the default metric for calculating the loss function in various scenarios, even though it might be slightly more challenging to interpret than MAE. These error measures of predictive accuracy are commonly employed to evaluate recommender models because they offer a straightforward and understandable way to assess performance.

B. *Evaluation is Based on Measuring the Model's Predicted Results*

An evaluation approach based on measuring the model's prediction results involves comparing the model's outcomes with the user's actual decisions to assess its effectiveness. This method employs information retrieval measures, using 2×2 confusion matrices, such as Precision, Recall, and F-score [13]. It is particularly well-suited for e-commerce applications as it aids in influencing users' decisions regarding product purchases or service usage. Two commonly used measures to evaluate the effectiveness of collaborative filtering models are Accuracy and Recall. However, certain models may exhibit contrasting values for Precision and Recall. In such cases, the F-score is employed to gauge the model's performance. A model is considered effective if these indicators exhibit high values [13]. By utilizing this evaluation method, recommender systems can be assessed based on their ability to align with users' preferences and improve their decision-making process, ultimately enhancing user satisfaction in e-commerce settings (see Table I).

TABLE I. CONFUSION MATRIX

	Relevant	Irrelevant	Total
Recommended	TP	FP	TP + FP
Not recommended	FN	TN	FN + TN
Total	TP + FN	FP + TN	N

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \tag{2}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \tag{3}$$

$$\text{F} - \text{score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision+Recall}} \tag{4}$$

In addition to the aforementioned evaluation measures, the precision-recall curve is utilized to compare the effectiveness of the proposed models against other models. The precision-recall curve is a straightforward graph that plots Precision values on the y-axis and Recall values on the x-axis. In other words, it represents TP/(TP+FN) on the y-axis and TP/(TP+FP) on the x-axis, both of which have values ranging from 0 to 1 based on the number of recommended items. Precision is also referred to as Positive Predictive Value, while Recall is known as Sensitivity, Hit Rate, or True Positive Rate. The ideal scenario for the algorithm is to achieve high values for both precision and recall. However, in many machine learning algorithms, there exists a trade-off between these two metrics, and optimizing one may come at the expense of the other. The quality of the precision-recall curve is determined by the area under the curve, with a larger area indicating a more effective model. This visual representation helps in comparing different models based on their performance in terms of precision and recall, providing valuable insights into the strengths and weaknesses of the proposed approaches.

VI. EXPERIMENT

A. *Experimental Datasets*

The proposed models undergo testing using two contextual rating datasets: De Paul Movie [14] and In Car Music. These datasets are organized based on the assumption that users share common ratings and contextual ratings. The De Paul Movie dataset comprises ratings provided by 97 students for 319 movies, with three context attributes considered: time, location, and companion. It consists of a total of 5043 rating values for movies across different contexts. For this dataset, three rating matrices are constructed, each corresponding to one of the context attributes. Each matrix contains 97 rows (representing the students) and 319 columns (representing the movies). On the other hand, the In Car Music dataset focuses on contextual music recommendations from a web service, gathered from 43 car drivers. This dataset includes 4012 rating values for 139 music songs based on eight context attributes: Driving Style, Landscape, Mood, Natural Phenomenon, Road Type, Sleepiness, Traffic Conditions, and Weather. Eight rating matrices are constructed for the In Car Music dataset, with each matrix having 43 rows (representing the drivers) and 139 columns (representing the music songs) to account for various contextual settings.

B. *Experimental Results*

1) *Compare accuracy based on predicted rating value*

The effectiveness of the proposed models is assessed by comparing their error parameters with two collaborative filtering models based on context similarity. The experiment involved deploying the content on two datasets across four models: the User-Based Multi-criteria Collaborative Filtering (MC-UBCF) model, the Item-Based Multi-criteria Collaborative Filtering (MC-IBCF) model, the Context-similarity User-context Collaborative Filtering (CUBCF) model, and the Context-similarity Item-Based Collaborative Filtering (CIBCF) model. Upon obtaining the experimental results, the error parameters (RMSE, MSE, and MAE) were calculated for each model based on each dataset. The comprehensive findings are presented in Table II, showcasing that the two multi-criteria collaborative filtering models outperformed the two context-similarity collaborative filtering models on both the De Paul Movie and In Car Music datasets. This demonstrates that the proposed multi-criteria models yielded improved error parameters, indicating their superiority in making accurate predictions and enhancing recommendation performance.

TABLE II. COMPARISON OF ERROR PARAMETERS OF MODELS ON TWO EXPERIMENTAL DATASETS

Datasets	Recommender model	RMSE	MSE	MAE
De Paul Movie	MC-UBCF	1.274960	1.625522	1.024712
	CUBCF	1.524717	2.324763	1.236806
	MC-IBCF	1.596529	2.548905	1.120736
	CIBCF	1.602497	2.567996	1.291709
In Car Music	MC-UBCF	1.540344	2.372661	1.259841
	CUBCF	1.685157	2.839755	1.334771
	MC-IBCF	1.902135	3.618117	1.456486
	CIBCF	1.819354	3.310048	1.617507

Specifically, the MC-UBCF model exhibits a significant reduction in error indices on both experimental datasets when compared to the CUBCF model (De Paul Movie ΔRMSE: 0.249757, ΔMSE: 0.699241, ΔMAE: 0.212094; In Car Music ΔRMSE: 0.144813, ΔMSE: 0.467094, ΔMAE: 0.074929). However, the MC-IBCF model only demonstrates a minimal reduction in error indices when compared to the CIBCF model on the De Paul Movie dataset (ΔRMSE: 0.005968, ΔMSE: 0.019091, ΔMAE: 0.170973). Notably, on the In Car Music dataset, the MC-IBCF model even shows higher error indices than the CIBCF model (ΔRMSE: −0.082781, ΔMSE: −0.308069). Based on the aforementioned comparison results, it is evident that the MC-UBCF model outperforms the MC-IBCF model on both experimental datasets. The MC-UBCF model demonstrates superior performance by achieving more substantial improvements in error metrics, showcasing its effectiveness in making accurate predictions and providing better recommendations.

2) *Compare accuracy based on prediction results*

In this evaluation method, the experimental content was tested using four models: MC-UBCF, MC-IBCF, CUBCF, and CIBCF, across two datasets. Precision, Recall, and F-score metrics were calculated from the prediction results of these models. The findings depicted in Fig. 3 reveal that the Precision value of the MC-UBCF model surpasses the CUBCF model on both experimental datasets. Conversely, the Recall index tends to exhibit the opposite trend. This indicates that the MC-UBCF model excels at making more accurate predictions when items are rated highly (positive). Additionally, the F-score index of the MC-UBCF model shows significant improvement compared to the CUBCF model on both experimental datasets, further underscoring the effectiveness of the MC-UBCF model over the CUBCF model. On the other hand, the experimental results for the MC-IBCF model indicate little notable improvement compared to the CIBCF model on both datasets. After comparing these outcomes, it is evident that the MC-UBCF model demonstrates the highest efficiency among the four experimental models. This highlights the strong performance of the user-based multi-criteria collaborative filtering model when applied to contextual rating data of users.

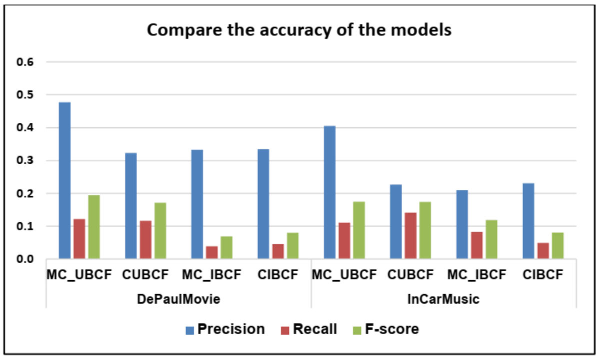

Figure 3. Compare the accuracy of the models on the experimental datasets.

3) *Compare accuracy based on precision-recall curve*

To further evaluate the performance of the two multi-criteria models on the experimental datasets, we generated precision-recall curves for these models and compared them with the similarity-based collaborative filtering models (CUBCF and CIBCF). The models were tested with varying numbers of items introduced to users, ranging from 1 to 10. Fig. 4 and Fig. 5 showcase the comparison of Precision/Recall ratios for the models on the two datasets. It is evident that the Precision/Recall ratio of the MC-UBCF model consistently outperforms the CUBCF model on both datasets. On the other hand, the Precision/Recall ratio of the MC-IBCF model does not show substantial improvement over the CIBCF model and is even lower on the In Car Music dataset. These findings reaffirm the effectiveness of the MC-UBCF model when dealing with contextual rating data from users. The precision-recall curves provide valuable insights into the models' performance in terms of precision and recall across different numbers of items introduced to users. The consistent superiority of the MC-UBCF model over the CUBCF model underscores its proficiency in making more accurate and reliable recommendations based on contextual data from users.

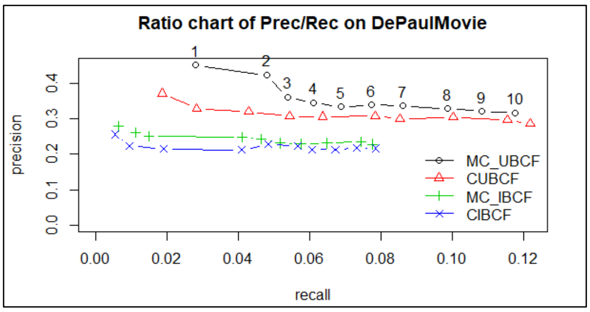

Figure 4. The chart compares the accuracy of models on De Paul Movie.

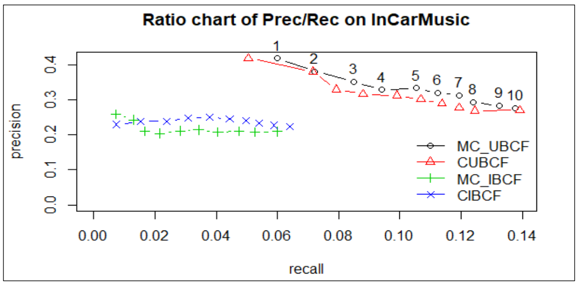

Figure 5. The chart compares the accuracy of models on In Car Music.

VII. Conclusion

This research endeavor developed two multi-criteria collaborative filtering models specifically designed for multi-context rating data, with the primary objective of enhancing the accuracy of collaborative filtering models. These models effectively segregate contextual user rating values into independent rating matrices, enabling personalized recommendations based on user contexts. Notably, the model considers each user's critical context, resulting in more precise predictions than those solely relying on overall rating values. Following the model's construction, it underwent testing on two sets of contextual data, and the obtained results were then compared with predictions based on overall rating data. The comparison-based results demonstrate that the multi-criteria collaborative filtering model outperforms the single-criteria collaborative filtering model, highlighting its practical application in e-commerce sites.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

All authors contributed equally to the conceptualization and design of the study. Nghia and Loi conducted the data collection and analysis. Nghia drafted the initial manuscript. Hiep provided critical revisions and contributed to the interpretation of results. All authors reviewed and approved the final version of the manuscript.

References

[1] L. Baltrunas *et al.*, "Incarmusic: Context-aware music recommendations in a car," *E-Commerce and Web Technologies, Springer Berlin Heidelberg*, pp. 89–100, 2011.

[2] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state of the art and possible extensions," *IEEE Transactions on Knowledge and Data Engineering*, pp. 734–749, 2005.

[3] G. Adomavicius and A. Tuzhilin, "Context-aware recommender system," *Recommender System Handbook*, pp. 217–253, 2011.

[4] F. Hdioud, B. Frikh, B. Ouhbi, and I. Khalil, "Multi-Criteria recommender systems: A survey and a method to learn new user's profile," *International Journal of Mobile Computing and Multimedia Communications*, vol. 8, no. 4, 29, 2017.

[5] H. X. Huynh, N. Q. Phan, N. M. Pham, V. H. Pham, L. H. Son, M. A. Basset, and M. Ismail, "Context-similarity collaborative filtering recommendation," *IEEE Access*, vol. 8, pp. 33342–33351, 2020.

[6] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Reidl, "Item-based collaborative filtering recommendation algorithms," in *Proc. 10th International Conference on World Wide Web*, pp. 285–295, 2001.

[7] M. Hassan and M. Hamada, "Smart media based context-aware recommender systems for learning: A conceptual framework," *Information Technology Based Higher Education and Training*, pp. 1–4, 2017.

[8] A. Kouadria, O. Nouali, and M. A. Shamri, "A multi-criteria collaborative filtering recommender system using learning-to-rank and rank aggregation," *Arabian Journal for Science and Engineering*, pp. 2835–2845, 2019.

[9] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: An open architecture for collaborative filtering of netnews," in *Proc. 1994 ACM Conference on Computer Supported Cooperative Work*, pp. 175–186, ACM, 1994.

[10] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: Itemto-item collaborative filtering," *IEEE Internet Computing*, vol. 7, no. 1, 2003.

[11] M. Hahsler, "Recommenderlab: A framework for developing and testing recommendation algorithms," *Information Retrieval*, 12371, 2022.

[12] F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, *Recommender Systems Handbook*, Springer Science + Business Media, pp. 1–34, 2011.

[13] C. C. Aggarwal, *Recommender Systems: The Textbook*, Springer International Publishing Switzerland, pp. 1–28, 2016.

[14] A. Odić, M. Tkalčič, J. F. Tasič, and A. Košir, *Predicting and Detecting the Relevant Contextual Information in a Movie Recommender System*, Oxford University Press, pp. 74–90, 2013.