https://doi.org/10.1186/s40537-024-00952-3

Journal of Big Data

RESEARCH Open Access

New custom rating for improving recommendation system performance

Tora Fahrudin[1*] and Dedy Rahman Wijaya[1]

*Correspondence:
Tora Fahrudin
torafahrudin@telkomuniversity.ac.id
[1]Lecturer at Telkom University, Bandung, Indonesia

Abstract
Recommendation system is currently attracting the interest of many explorers. Various new businesses have surfaced with the rise of online marketing (E-Commerce) in response to Covid-19 pandemic. This phenomenon allows recommendation items through a system called Collaborative Filtering (CF), aiming to improve shopping experience of users. Typically, the effectiveness of CF relies on the precise identification of similar profile users by similarity algorithms. Traditional similarity measures are based on the user-item rating matrix. Approximately, four custom ratings (CR) were used along with a new rating formula, termed New Custom Rating (NCR), derived from the popularity of users and items in addition to the original rating. Specifically, NCR optimized recommendation system performance by using the popularity of users and items to determine new ratings value, rather than solely relying on the original rating. Additionally, the formulas improved the representativeness of the new rating values and the accuracy of similarity algorithm calculations. Consequently, the increased accuracy of recommendation system was achieved. The implementation of NCR across four CR algorithms and recommendation system using five public datasets was examined. Consequently, the experimental results showed that NCR significantly increased recommendation system accuracy, as evidenced by reductions in RMSE, MSE, and MAE as well as increasing FCP and Hit Rate. Moreover, by combining the popularity of users and items into rating calculations, NCR improved the accuracy of various recommendation system algorithms reducing RMSE, MSE, and MAE up to 62.10%, 53.62%, 65.97%, respectively, while also increasing FCP and Hit Rate up to 11.89% and 31.42%, respectively.

Index terms Recommendation system, Collaborative filtering, New rating formula, E-commerce

Introduction

Several research are currently deploying recommendation system in many fields such as E-Commerce [1], social networks [2], news [3], online services, search engine [4], streaming services [5], healthcare service recommendation [6], and restaurant services [7]. This widespread adoption is due to the abundance of both explicit and implicit user data available across the fields. Examples of the data include social media data, wearable sensors, medical data, smart device usage, click data of user, and behavior patterns of user. Typically, all of the data can be leveraged to build an effective recommendation system. There are three general methods of building recommendation system models namely, Content-Based Filtering (CBF), Collaborative Filtering (CF), and Hybrid System (HS) [8].

CBF is a method for recommending items based on the similarity of description or features to other profiles of user [9]. The system provides recommendations by analyzing the history of users' past preferences in this method. To learn these preferences, recommendation system uses some machine learning algorithms such as Decision Tree [10], Naive Bayes classifier [11], SVM [12], and Deep Learning [13]. Additionally, CBF offers two advantages which include the model does not require any data from users and the model can recommend items with fewer fans. However, the model has the following disadvantages such as the dependency of its accuracy based on sufficiency of information which requires intensive domain knowledge. The model also has limited ability to expand on the users' existing interests [14].

CF is recommendation system method that uses similarity algorithms to find a user with preferences similar to the neighborhood [9]. Furthermore, it uses interactions between users and items, such as ratings to make recommendations [15]. CF is divided into two methods namely model-based and memory-based [16]. Specifically, memory-based can be classified into item-based and user-based CF methods. The difference between model and memory is that a memory-based algorithm loads the entire database into system memory to make recommendation while a model-based algorithm compresses large database into a model and performs recommendation task using reference mechanism [17]. In model-based, latent factors are used to predict preference score of undetected items. [18]. The method includes two steps, namely computing matrix factorization, followed by generating recommendation based on machine learning algorithms. The advantage of CF is that it can recommend hidden items to users. However, CF method also has disadvantages such as data scarcity, cold-start problem, and scalability issues.

HS is recommendation system that combine two or more methods to obtain better performance by eliminating the drawbacks of each method [14]. Some of the combination strategies include combining CF with Content-Based method and Self-Organizing Map neural network method [19], combining CF system using SVD algorithm with a content-based system, and a fuzzy expert system [20]. Moreover, other strategies include joining CF to make recommendations based on information equations between users and CBF [21]. In general, there are seven hybridization mechanisms used to build hybrid recommendation system including weighted, mixed, switching, feature combination, feature augmentation, cascade, and meta-level [22].

Among the methods mentioned, user and item ratings are two major data sources used by recommendation system [23]. The degree of preference in recommendation system is measured by a rating score [24]. Item recommendation is usually sorted by predicted rating in descending order. Typically, items rating are represented as User, Item, and Rating [25] while CF method often uses average rating. This method assumes that rating value for all items is the same, without considering the number of users. Custom rating (CR) can be used to determine final rating when considering the popularity of users and items to determine new ratings value. Based on available data, the current research is the first to explore CR with the aim of improving performance of recommendation system in some public dataset.

Related work

Improving rating formulas to increase recommendation system accuracy is an interesting idea. According to [26], rating is a number that expresses a user's opinion about a given product. It is usually between 1 and 5, with higher values showing better ratings. To generate recommendations, similarity algorithms are used on the row or column vectors of 'user x product' matrix of ratings. Moreover, this recommendation is built by finding the closest user's profile based on the similarity with other users.

Diverse and popular E-Commerce applications, such as books, movies, products, and videos, frequently include a rating feature. This rating is a major source of data used by recommendation system [23]. Some results have explored customized rating schemes in the system including Weighted Average Ratings, which are used in IMDb rating [27], Bayesian Average rating [28], Median Average Rating [29], Mean Rating [30], and Normalized Rating Frequency [31]. Additionally, research has derived rating values from contextual data such as aspect category [32], word embedding representation of reviews [23], and Linked Open Data [33]. It is crucial to be aware that recommendation system has the main task of achieving rating prediction [34]. Therefore, this research focuses on exploring rating prediction method across dataset of movie, music, and jokes signifying the accuracy of this prediction.

Rating in recommendation system

Rating is one of the important features of recommendation system. The main function of rating is to determine a user's preference or interest in a product. Typically, users can like or dislike preferred item. This rating helps recommendation system determine suitable items for users. In the context of this research, rating can be considered explicit or implicit [35]. Specifically, explicit rating is given directly by users through a scale or rating system such as 1–5 or 1–10 [36]. Meanwhile, implicit rating is derived from user interactions with a product such as viewing or clicking on the item [37].

Apart from being used to determine recommendations, ratings can also be used in recommendation system for other purposes, including:

- Evaluation of recommendation system: Ratings can be used to evaluate the accuracy of recommendation system [9] by comparing user-given to those predicted by the system.
- Improved accuracy: A new rating is used to improve the accuracy of recommendation system [38].
- Personalization of recommendations: Ratings can be used to create more personalized recommendations [39].
- Pricing: Ratings are used to determine the price of an item [40].
- Compared to implicit rating, explicit rating has many advantages as follows:
- Accuracy: Ratings provided by users are explicitly more accurate than those derived from user interactions with items. In explicit rating, user directly shows the level of preference for an item, which is particularly reliable when reviewers possess a high degree of expertise [41].

- Specificity: Explicit ratings provides more specific information about user preferences than implicit ratings. Users can give different ratings to various items, thereby providing more specific information about preferences.
- More data: Implicit ratings are often derived from limited data such as clicks or other interactions. Meanwhile, explicit ratings typically come from more data, such as comments or reviews [42].
- Transparency: Since user explicitly provides rating for item, this rating is more transparent and easier to understand than an implicit rating obtained from interactions that may not be clearly understood.

The choice of method depends on the application's conditions and the availability of the necessary data. Explicit ratings can provide more comprehensive and valid data but require higher participation from users. Meanwhile, implicit rating is easier to implement because it does not require active user contribution. However, the data obtained from implicit rating is less specific.

Custom rating

CR is made to accommodate specific needs, criteria, and contexts. The contexts can vary for each type of recommendation system environment and can be changed according to user preferences or business requirements. For example, in movie recommendation system, CR such as Weighted Average Rating (WR) calculates rating by considering the number of votes, average rating of item, minimum votes required to be listed, and the mean vote. Another CR method uses median rating, which takes the middle value of a set of rating data. Furthermore, the mean rating or average rating shows the total average level of an item or product. Lastly, Normalized Rating Frequency (NRF) uses rating frequency and applied normalization process.

Weighted average rating

The purpose of rating is to address the skewness problem that often occurs in system. This method helps to manage inconsistencies in film ratings which can change as the number of voters increases. For example, a newly released film with only a small number of voters may have a very high average rating because it is only judged by very enthusiastic viewers. However, as the film reaches broader audience over time, the average rating may drop and become more accurate.

$$Weighted\,Rating\,(WR) = \left(\frac{v}{v+m}R\right) + \left(\frac{m}{v+m}C\right) \tag{1}$$

Where v is the number of votes for the movie, m represents minimum votes required to be listed in the chart, R is the average rating of the movie, and C is the mean vote across the whole report.

Median rating

The median rating is the middle value of a set of rating data. typically, the median is determined by sorting all the rating data and selecting the middle value. When the number of rating data is odd, the median is the middle, when even, it is the average of the two middle values. Furthermore, the median rating is often used in recommendation system to address data skewness problems. For example, a film with a small number of high ratings may have a very high average rating which may not represent the views of most viewers. The median rating can provide a more accurate picture, reflecting a central tendency that is less affected by extreme values.

$$Median\left(X\right)=\begin{cases} X\left[\frac{n+1}{2}\right] & if\, n\, is\, odd \\ \frac{X\left[\frac{n}{2}\right]+X\left[\frac{n}{2}+1\right]}{2} & if\, n\, is\, even \end{cases} \tag{2}$$

Mean rating

The mean rating or average rating shows the average level of an item or product rated by several people. This value provides an understanding of the total rating of an item or product. However, the mean rating has weaknesses, such as being affected by very high or very low rating values from one or several individuals. Therefore, other methods such as the median rating mode are used to determine the average level of an item or product more accurately.

$$\bar{x}=\frac{\sum x_i}{n} \tag{3}$$

Normalized Rating Frequency (NRF)

NRF is a user-rating method focused on normalizing rating frequency [31]. The normalization process is based on calculating the frequency of user rating, eventually leading to product ranking for recommendation. In addition, NRF method consists of 5 stages as follows:

- Calculating Rating Frequency: The frequency is calculated by summing each user rating occurrence. u presents list of users, p, list of products, and r presents list of user-rating.

$$MF_{(r_i,p_h)}=\sum_{u=1}^{l}\begin{cases} 1\rightarrow if M\left(u_g,p_h\right)=r_i \\ 0\rightarrow if\ Otherwise \end{cases} \tag{4}$$

- Calculating Total Frequency Rating: The total frequency rating is obtained by summing $M\left(u_a,p_h\right)$ for each product. This frequency is presented by $Total_{(ri)}$.

$$Total_{(r_i)}=\sum_{p} MF\left(r_i,p_h\right) \tag{5}$$

- Normalization: The normalization guarantees that $MF\left(r_i,p_h\right)$ is scaled from 0 to 1. Additionally, this normalization can overcome the differences in scale of units in rating frequency. The normalization process is presented by $N\left(r_i,p_h\right)$.

$$N\left(r_{i,}p_h\right)=\begin{cases} 0\rightarrow if\ Total_{(r_i)} \\ \frac{MF(r_i,p_h)}{Total_{(r_i)}}\rightarrow if\ Otherwise \end{cases} \tag{6}$$

- Weighted Rating: Weighted rating $(NR(r_i, p_h))$ helps to differentiate rank values among products by avoiding identical rating values.

$$NR\left(r_i, p_h\right) = N\left(r_i, p_h\right) * r_i \tag{7}$$

- Total Weighted Rating: Total Weighted Rating represents the sum of all weighted rating each product. In addition, the rating is formulated as follows:

$$NRF\left(p_h\right) = \sum\nolimits_r NR\left(r_i, p_h\right) \tag{8}$$

Proposed method

The section discussed a new method to improve recommendation system by using four CR combined with new rating formula that joined frequency count and the original rating. Generally, the method aimed to ensure the effectiveness of items rating by considering users' popularity through the frequency of given ratings, original users rating, and items popularity through CR. The proposed method, called NCR was shown in following formula:

$$Ncr_{(r_i, p_h)} = \sqrt[3]{F_{U_i} * CR_{P_h} * r_i} \tag{9}$$

Where F_{U_i} presents normalized and scaled frequency count of *i*-user from all products rated by *i*-user. CR_{P_h} represents CR for *h*-product and r_i represents user-rating for *h*-product respectively. Furthermore, CR was achieved using weighted average rating, median, mean, and normalized rating frequency in this experiment. F_{U_i}, CR_{P_h}, and r_i were scaled to the same range, such as 1–5. Additionally, Fig. 1 showed the experimental design of NCR method.

Four CR such as Weighted Average Rating, Median, Mean, and Normalized Rating Frequency were used. For the experimental research, five public datasets were used which included MovieLens (100 K), MovieLens (1 M), MovieLens Latest Small (2 K), Jester, and Last.fm. Furthermore, Table 1 showed the number of users, rating, product, and density for each dataset. All dataset rating was normalized in 1 to 5 scaled. In system recommendations, "data density" referred to the degree to which user-item data in a dataset was populated. Data density was the ratio between the number of entries or ratings in the dataset and the theoretically possible number of entries. Additionally, higher-density data meant there were more entries or ratings in the dataset, providing more complete and relevant information for making recommendations. The following process led to more accurate and reliable recommendations. Moreover, low data density, showed many entries or ratings were missing, making it difficult to find patterns or sufficient information to provide relevant recommendations. Moreover, low data density also led to problems in training and testing recommendation models, due to a shortage of available data.

Experimental tests were conducted on an Intel (R) Core i7-9750 H machine with 2.6 GHz clock frequency and 8 GB of RAM, running on Windows 11 platform with Python programming language. Additionally, a 10-fold cross-validation procedure with the same Random_State (RS) was used. Typically, RS parameter referred to an integer

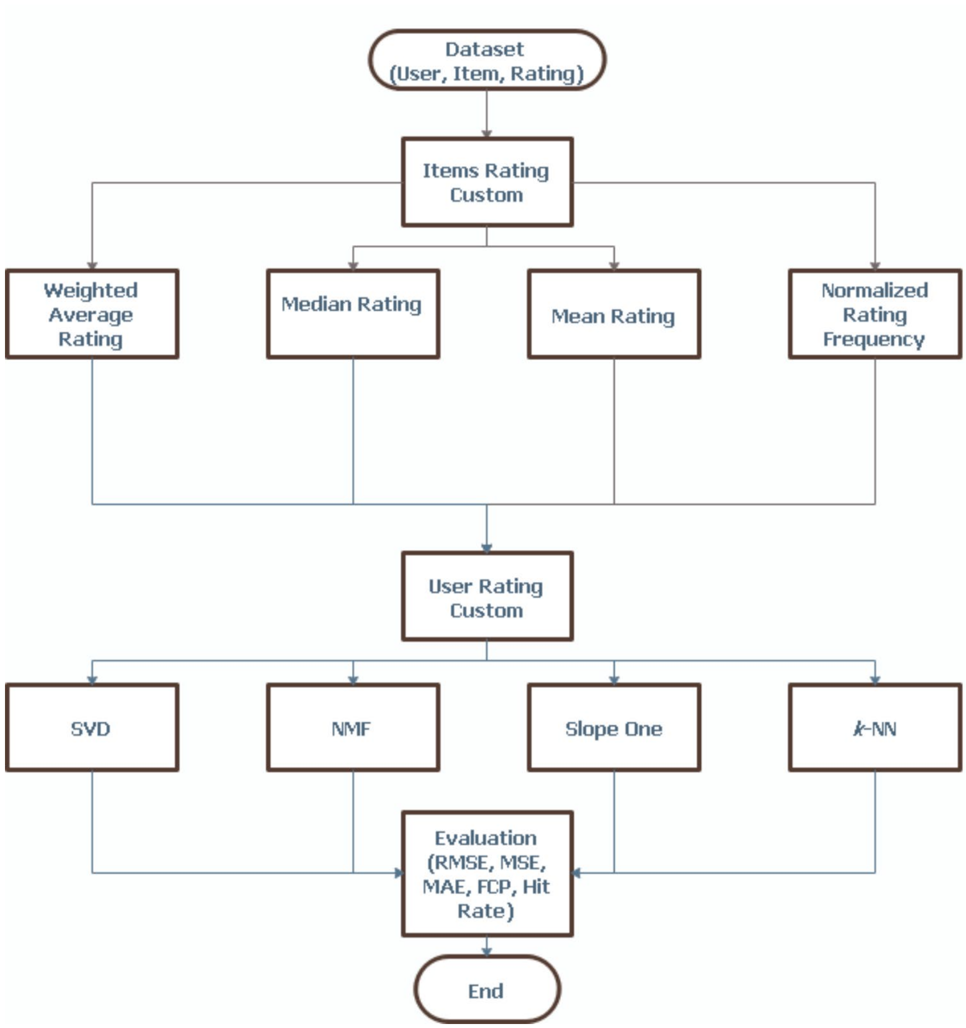

Fig. 1 Experimental design of NCR

Table 1 Dataset description

No	Dataset	Users	Rating	Product	Density
1	Movie Lens (100k)	943	1,000,000	1682	6.3%
2	Movie Lens (1 M)	6040	1,000,209	3706	4.4%
3	Movie Lens (Latest Small)	610	100,836	9724	1.6%
4	Jester	51,932	1,761,439	140	21.3%
5	Last.fm (2 K)	1892	92,834	17,632	0.28%

or a random generator object. When the value was provided, the data division remained consistent every time cross-validation was performed with the same RS parameter. This process was useful for ensuring consistent and reproducible results when using cross validation. Subsequently, an average of all the results was compared across all algorithm and scenario. To evaluate the performance of RS, different similarity measure was used each time. Therefore, the performance of recommendation system using four well-known evaluation metrics was evaluated as follows.

- Root Mean Squared Error (RMSE): RMSE measured the differences between predicted values and the observed values, assessing the quality of fit between the actual data and the model's prediction. Additionally, RMSE was one of the most frequently used metrics for evaluating the goodness of fit of generalized regression models. RMSE had several advantages including error representation on the same scale, sensitivity to large errors, and more stability to scale changes.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \widehat{y}_i\right)^2} \tag{10}$$

- Mean Squared Error (MSE): An evaluation metric commonly used in recommendation system that measured the difference between the predicted value and the actual value in terms of rating or user preference for an item. MSE was calculated by taking the squared difference between the predicted and actual value for each rating or user preference for the item, followed by averaging the squared difference. Furthermore, lower MSE value showed better quality predictions from recommendation system. Using MSE for recommendation system metrics had several advantages such as better error representation and sensitivity to value differences.

$$MSE = \frac{1}{n}\sum_{i}^{n}\left(y_i - \widehat{y}_i\right)^2 \tag{11}$$

- Fraction of Concordant Pairs (FCP): The objective of FCP was to measure the degree of consistency or agreement between two ordered data sets. Typically, these metrics were used to compare two measurement methods or two explorers assessing the same item. FCP used pairs of ratings from the test set of the data. Each pair consisted of two rated items from the same user, with every possible combination of items rated by that user forming a pair. Following this process, a higher FCP value showed greater accuracy of the algorithm. For example, when FCP value was 0.5, it meant that 50% of the item pairs being compared had a concordant ranking order. However, when FCP value was 0.8, then 80% of the item pairs being compared had a concordant ranking order. In both cases, higher FCP score showed better quality recommendation system.

$$FCP = \frac{n_c}{n_c + n_d} \tag{12}$$

- Hit Rate: Hit Rate was the fraction of users for whom the correct answer was included in recommendation list of length L. $\left|U_{hit}^{L}\right|$ represented the number of users for whom the correct answer was included in the top L recommendation list, while $\left|U_{all}\right|$ was the total number of users in the test dataset. Furthermore, this method evaluated the quality of recommendation based on the success in predicting the items a user was genuinely interested in, considering the relevance of recommendations provided by the system. Hit Rate was typically expressed as a value between 0 and 1, with higher value showing better performance in recommendation system. For example, when Hit Rate of 0.6 with L = 5, it meant that the system successfully recommended at least one correct item in the top 5 items by 60% of the users evaluated.

$$HitRate = \frac{\left|U_{hit}^{L}\right|}{\left|U_{all}\right|} \tag{13}$$

Experimental result

Results were presented by applying a formula, which used weighted average rating, median rating, mean rating, and NRF scheme in this section. Additionally, the result was compared with the original rating data. The experimental exploration was conducted using the same datasets, 10-fold cross-validation, and the same random state. The top 10 recommendation settings were used for each user from a set of predictions.

RMSE as a measure

The result was analyzed based on RMSE as a measure in this sub section. Table 2 showed the results for RMSE metrics obtained for the four algorithms across five dataset, showing the best results in bold for each dataset. The Table showed that using a new formula rating derived from the four-rating method decreased RMSE value by approximately 62.10% compared to the original rating. Subsequently, introducing a new formula using NRF as CR method achieved the smallest RMSE in three datasets compared to the other CR methods. By conducting the Friedman Test, a non-parametric statistical test aimed at determining whether there was a significant difference between the groups. For example, whether there was at least one group that consistently ranks higher or lower than the

Table 2 RMSE results

Word	Dataset	Algorithms	Original	Weighted average rating	Median rating	Mean rating	Normalized rating frequency
1	Movie Lens (100k)	SVD	0.93	0.301	0.314	0.309	0.156
		NMF	0.956	0.298	0.304	0.302	**0.144**
		Slope One	0.942	0.299	0.306	0.303	0.173
		k-NN (user-based)	0.972	0.331	0.337	0.335	0.146
		k-NN (item-based)	1.02	0.351	0.411	0.393	0.224
2	Movie Lens (1 M)	SVD	0.866	0.265	0.268	0.266	0.083
		NMF	0.914	0.268	0.271	0.269	0.092
		Slope One	0.905	0.268	0.27	0.269	0.118
		k-NN (user-based)	0.917	0.278	0.28	0.278	**0.079**
		k-NN (item-based)	0.997	0.323	0.379	0.36	0.137
3	Movie Lens (Latest Small)	SVD	0.773	0.288	0.284	0.279	0.31
		NMF	0.814	**0.252**	0.268	0.267	0.359
		Slope One	0.806	0.272	0.291	0.289	0.415
		k-NN (user-based)	0.836	0.397	0.409	0.407	0.376
		k-NN (item-based)	0.87	0.319	0.392	0.376	0.606
4	Jester	SVD	0.821	0.313	0.314	0.312	0.184
		NMF	0.855	0.301	0.303	0.301	0.118
		Slope One	0.847	0.295	0.297	0.295	0.149
		k-NN (user-based)	0.847	0.351	0.352	0.35	**0.115**
		k-NN (item-based)	0.897	0.332	0.361	0.351	0.162
5	Last.fm (2 K)	SVD	0.066	0.075	0.075	0.075	0.521
		NMF	0.042	0.062	0.062	0.063	0.689
		Slope One	0.04	**0.024**	**0.024**	**0.024**	0.862
		k-NN (user-based)	0.041	0.035	0.035	0.035	0.689
		k-NN (item-based)	0.047	0.026	0.025	0.026	0.918

other, a test statistic of 42.589 and corresponding p-value of $p=0.00$ were obtained. This showed statistically significant differences in RMSE from all rating methods. Compared to the original rating, all CR formulas showed performance improvement by decreasing RMSE results in four datasets across all algorithms. Additionally, weighted average rating showed the best performance compared to the other CR algorithms.

MSE as a measure

MSE was used as a measure to assess CR performance in this sub section. Table 3 showed that using new formula rating derived from the four-rating method decreased the average MSE value by approximately 53.62% compared to the original rating. Subsequently, NRF outperformed the others in achieving the smallest MSE value, as evidence by its performance across three datasets. The test produced a statistic of 37.96 and corresponding p-value of $p=0.00$, showing statistically significant differences in MSE among all rating method.

MAE as a measure

Similar to RMSE and MSE, MAE served as a measure where CR significantly outperformed the original rating. Table 4 showed the results for MAE metrics obtained across four algorithms and five datasets, with best values showed in bold. According to Table III the use of a new formula derived from the four-rating method could improve performance by decreasing the MAE value by approximately 65.97% compared to the original

Table 3 MSE results

No	Dataset	Algorithms	Original	Weighted average rating	Median rating	Mean rating	Normalized rating frequency
1	Movie Lens (100k)	SVD	0.865	0.09	0.098	0.096	0.024
		NMF	0.915	0.089	0.092	0.091	**0.02**
		Slope One	0.887	0.089	0.093	0.092	0.03
		k-NN (user-based)	0.945	0.109	0.114	0.112	0.021
		k-NN (item-based)	1.041	0.123	0.17	0.155	0.05
2	Movie Lens (1 M)	SVD	0.75	0.07	0.072	0.071	**0.006**
		NMF	0.835	0.072	0.072	0.073	0.008
		Slope One	0.82	0.071	0.073	0.072	0.013
		k-NN (user-based)	0.841	0.077	0.078	0.077	**0.006**
		k-NN (item-based)	0.995	0.104	0.143	0.13	0.018
3	Movie Lens (Latest Small)	SVD	0.598	0.083	0.08	0.078	0.096
		NMF	0.663	**0.063**	0.072	0.071	0.129
		Slope One	0.651	0.074	0.085	0.083	0.172
		k-NN (user-based)	0.7	0.157	0.167	0.165	0.142
		k-NN (item-based)	0.758	0.102	0.154	0.141	0.367
4	Jester	SVD	0.674	0.098	0.098	0.097	0.033
		NMF	0.732	0.091	0.092	0.09	0.014
		Slope One	0.718	0.087	0.088	0.087	0.022
		k-NN (user-based)	0.717	0.123	0.124	0.122	**0.013**
		k-NN (item-based)	0.804	0.11	0.13	0.123	0.026
5	Last.fm (2 K)	SVD	0.004	0.005	0.005	0.005	0.271
		NMF	**0.001**	0.003	0.003	0.003	0.474
		Slope One	**0.001**	**0.001**	**0.001**	**0.001**	0.744
		k-NN (user-based)	**0.001**	**0.001**	**0.001**	**0.001**	0.475
		k-NN (item-based)	0.002	**0.001**	**0.001**	**0.001**	0.842

Table 4 MAE results

No	Dataset	Algorithms	Original	Weighted average rating	Median rating	Mean rating	Normalized rating frequency
1	Movie Lens (100k)	SVD	0.732	0.226	0.234	0.231	0.098
		NMF	0.752	0.222	0.226	0.224	0.098
		Slope One	0.74	0.224	0.228	0.227	0.12
		k-NN (user-based)	0.767	0.251	0.255	0.253	**0.091**
		k-NN (item-based)	0.804	0.264	0.308	0.294	0.142
2	Movie Lens (1 M)	SVD	0.679	0.201	0.204	0.202	**0.052**
		NMF	0.722	0.2	0.202	0.201	0.072
		Slope One	0.713	0.201	0.203	0.202	0.084
		k-NN (user-based)	0.722	0.211	0.213	0.211	**0.052**
		k-NN (item-based)	0.777	0.238	0.279	0.265	0.081
3	Movie Lens (Latest Small)	SVD	0.593	0.207	0.201	0.199	0.181
		NMF	0.625	0.181	0.187	0.186	**0.163**
		Slope One	0.615	0.198	0.207	0.205	0.236
		k-NN (user-based)	0.641	0.286	0.292	0.291	0.181
		k-NN (item-based)	0.678	0.242	0.292	0.28	0.442
4	Jester	SVD	0.623	0.236	0.237	0.236	0.137
		NMF	0.647	0.222	0.224	0.222	0.093
		Slope One	0.644	0.217	0.219	0.218	0.116
		k-NN (user-based)	0.634	0.259	0.261	0.259	**0.079**
		k-NN (item-based)	0.67	0.24	0.264	0.256	0.118
5	Last.fm (2 K)	SVD	0.033	0.055	0.054	0.055	0.328
		NMF	**0.008**	0.047	0.046	0.047	0.299
		Slope One	**0.007**	**0.006**	**0.005**	**0.006**	0.627
		k-NN (user-based)	**0.008**	**0.007**	**0.007**	**0.008**	0.254
		k-NN (item-based)	0.009	0.007	0.005	0.007	0.657

rating. Additionally, a new formula using NRF as CR method achieved the smallest MAE in four datasets compared to the other CR methods. By conducting the Friedman Test, test statistic of 36.391 was obtained and the corresponding p-value of $p=0.00$, showing statistically significant differences in MAE from all rating method.

FCP as a measure

Similar to RMSE and MSE, in terms of MAE as a measure, CR outperformed the original rating significantly. Table 5 showed the results for FCP metrics obtained across four algorithms and five dataset, with best values showed in bold. Therefore, it was concluded that the new formula derived from the four-rating method improved performance by about 11.89% compared to the original rating. Moreover, a new formula using NRF as CR method achieved the smallest RMSE in four datasets compared to the other CR methods. By conducting the Friedman Test, test statistic of 45.768 and a corresponding p-value of $p=0.00$ was obtained, showing statistically significant differences in FCP among all rating method.

Hit rate as a measure

Similar to other evaluation metrics mentioned earlier, Hit Rate for CR was 31.42% higher than standard rating as shown in Table 6. NRF algorithms appeared as the best-performance algorithms across four dataset. According to Friedman Test, the explorers obtained a test statistic of 26.505 and corresponding p-value of $p=0.00$,

Table 5 FCP results

No	Dataset	Algorithms	Original	Weighted average rating	Median rating	Mean rating	Normalized rating frequency
1	Movie Lens (100k)	SVD	0.691	0.723	0.747	0.745	0.726
		NMF	0.679	0.728	0.752	0.75	0.75
		Slope One	0.683	0.735	0.755	0.756	0.624
		k-NN (user-based)	0.697	0.729	**0.758**	0.749	0.752
		k-NN (item-based)	0.598	0.614	0.613	0.604	0.608
2	Movie Lens (1 M)	SVD	0.739	0.763	0.792	0.786	0.774
		NMF	0.71	0.757	0.785	0.78	0.777
		Slope One	0.715	0.764	0.792	0.787	0.571
		k-NN (user-based)	0.727	0.763	**0.797**	0.783	0.781
		k-NN (item-based)	0.667	0.681	0.698	0.678	0.707
3	Movie Lens (Latest Small)	SVD	0.653	0.707	0.734	0.721	**0.795**
		NMF	0.643	0.713	0.755	0.739	0.767
		Slope One	0.653	0.699	0.741	0.727	0.704
		k-NN (user-based)	0.664	0.675	0.724	0.709	0.789
		k-NN (item-based)	0.432	0.458	0.444	0.465	0.223
4	Jester	SVD	0.593	0.621	0.664	0.66	0.677
		NMF	0.573	0.607	0.65	0.647	0.682
		Slope One	0.575	0.626	0.668	0.665	0.614
		k-NN (user-based)	0.586	0.534	0.593	0.587	**0.687**
		k-NN (item-based)	0.578	0.597	0.628	0.626	0.676
5	Last.fm (2 K)	SVD	0.42	0.502	0.494	0.501	**0.793**
		NMF	0.228	0.555	0.508	0.543	0.758
		Slope One	0.318	0.373	0.356	0.373	0.64
		k-NN (user-based)	0.504	0.693	0.612	0.688	0.74
		k-NN (item-based)	0.456	0.431	0.441	0.431	0.43

showing statistically significant differences among all rating methods in terms of Hit Rate achievements.

Rating and custom rating in boxplot

Box plots provided an overview of data distribution, range, mean, diversity, and the presence of outlier values. The explorers enabled comparison between the new distribution of CR data and the original rating data. The length of the box showed the data's spread, with a longer box showing greater dispersion. In a vertical boxplot, the lines below the boxes represented the lower whiskers, while those above represented the upper whiskers. Additionally, values beyond the whisker were termed outliers and extremes.

Figure 2 showed the distribution of both the original rating and CR data in Movie Lens 100 K. The explorers observed that the rating data distribution changed after modifications. All rating modification algorithms showed smaller median data compared to the original rating, according to the boxes positioned beneath those of the original data. Furthermore, the quartile range became shorter while the outlier data was greater. This showed a significant alteration in the rating value due to the combination of the popularity of users and items with the original rating. Some outlier values were present in all CR algorithms, evident in values above Q3 + (1.5 x IQR) and below Q1 - (1.5 x IQR) showing the influence of ratings frequency (F), original users rating (r), and items popularity through CR in NCR formula on the new rating distribution. Although some of new

Table 6 Hit rate results

No	Dataset	Algorithms	Original	Weighted average rating	Median rating	Mean rating	Normal-ized rating frequency
1	Movie Lens (100k)	SVD	0.208	0.405	**0.467**	0.447	0.415
		NMF	0.205	0.408	0.416	0.45	0.334
		Slope One	0.208	0.406	0.416	0.414	0.39
		k-NN (user-based)	0.208	0.372	0.383	0.384	0.384
		k-NN (item-based)	0.172	0.408	0.399	0.405	0.396
2	Movie Lens (1 M)	SVD	0.264	0.369	0.383	0.381	**0.418**
		NMF	0.231	0.383	0.397	0.395	0.283
		Slope One	0.242	0.382	0.394	0.394	0.394
		k-NN (user-based)	0.274	0.34	0.358	0.358	0.391
		k-NN (item-based)	0.308	0.406	0.408	0.413	0.309
3	Movie Lens (Latest Small)	SVD	0.356	0.395	0.409	0.409	**0.47**
		NMF	0.343	0.447	0.416	0.415	0.427
		Slope One	0.361	0.442	0.414	0.412	0.413
		k-NN (user-based)	0.37	0.418	0.382	0.377	0.435
		k-NN (item-based)	0.318	0.365	0.315	0.321	0.432
4	Jester	SVD	0.328	0.417	0.421	0.419	0.444
		NMF	0.33	0.448	0.446	0.392	0.05
		Slope One	0.333	0.446	0.446	0.446	0.399
		k-NN (user-based)	0.344	0.442	0.444	0.445	**0.462**
		k-NN (item-based)	0.342	0.443	0.439	0.44	0.395
5	Last.fm (2 K)	SVD	0.286	0.282	0.275	0.282	0.494
		NMF	0.05	0.052	0.056	0.053	0.487
		Slope One	0.46	0.43	0.423	0.428	0.468
		k-NN (user-based)	0.446	0.425	0.428	0.43	**0.498**
		k-NN (item-based)	0.471	0.47	0.459	0.47	0.245

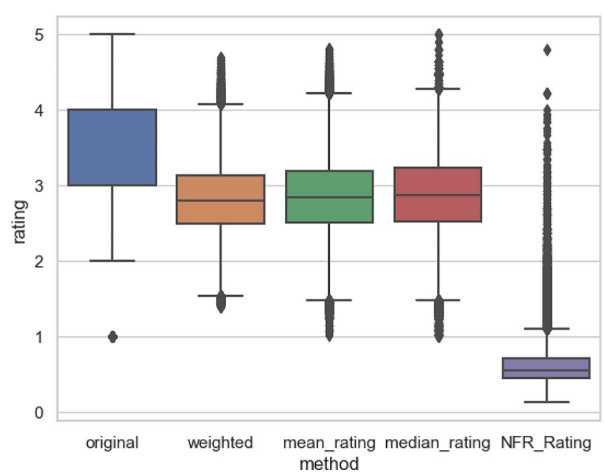

Fig. 2 Box plot movie lens 100k

rating was located in outlier area due to the effect of item popularity, this phenomenon increased the accuracy of recommendation system.

Figure 3 showed the distribution of both the original rating and CR data in Movie Lens 1 M. Similar to Fig. 2, explores observed a change in the rating data distribution after modifications. The median rating value as shown by the boxplot was lower compared to the original rating. Additionally, interquartile range had shrunk leading to a narrower

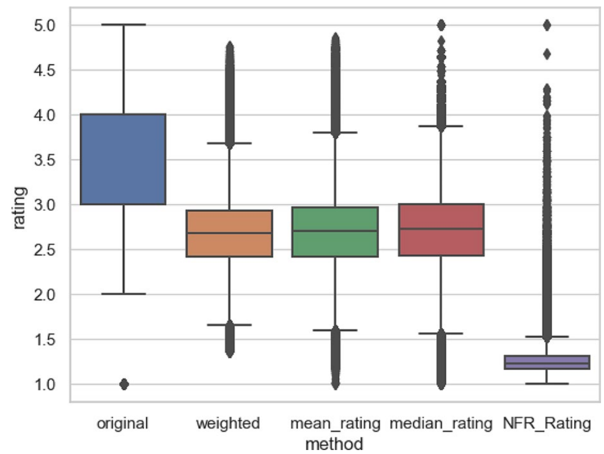

Fig. 3 Box plot movie lens 1 M

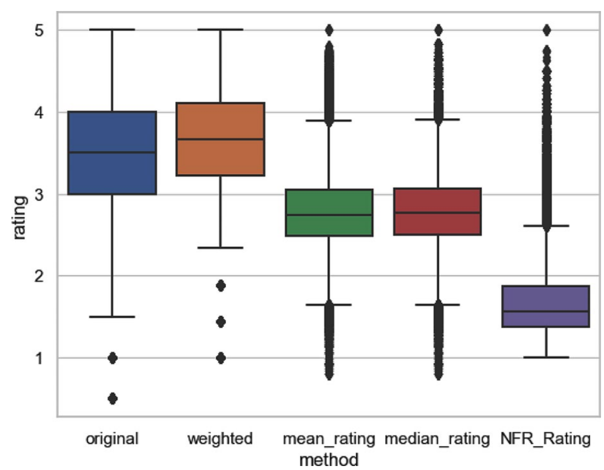

Fig. 4 Box plot movie lens (latest small)

data distribution. Among custom algorithms, NFR rating algorithms showed the smallest interquartile range.

Figure 4 showed different distribution with the median data in weighted average rating exceeding the original. The length of the box and whisker of this rating was smaller than the original. In comparison to both the original and weighted average rating, the mean, median, and NFR rating showed lower median and numerous outlier values.

Compared to movie lens dataset in jester dataset shown in Fig. 5, the new rating showed symmetrical data distribution. Interquartile length of weighted, mean, and median rating was smaller than the original, showing reduced variability in the distribution of new rating data. Subsequently, an outlier value in NFR rating significantly contributed to improving performance.

Figure 6 finally showed that the length of NFR rating interquartile was significantly greater than that of the others. Median value of all new CR algorithms was almost the same. Additionally, the spread of new rating data from weighted, mean, and median rating was also almost the same distribution.

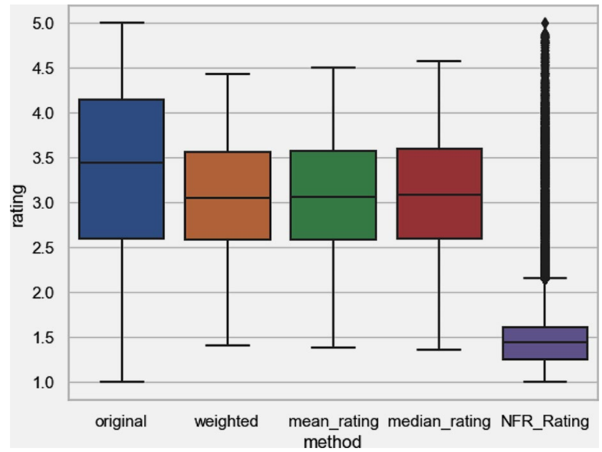

Fig. 5 Box plot jester

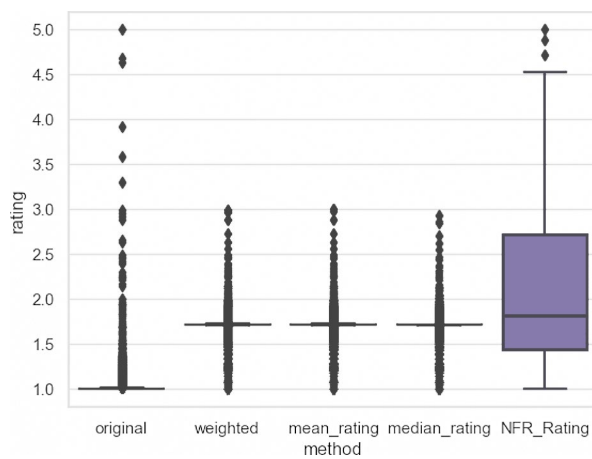

Fig. 6 Box plot last fm

Result and discussion

The results were represented where the original rating were compared with those obtained using CR. The experimental research was conducted using the same datasets, RS parameters, and algorithms. Furthermore, the results from five evaluation parameters showed that CR could increase recommendation system's accuracy by decreasing RMSE, MSE, and MAE while increasing FCP and Hit Rate. Various CR methods were applied and the accuracy was assessed across five public datasets. The experiment results led to the conclusion that combining the popularity of users and items with the original rating improved recommendation. This combination altered the distribution of the new rating data, as observed from the boxplot results. Additionally, the result was evident that the inclusion of users and items popularity improved recommendation performance, as shown by the new RMSE, MSE, MAE, FCP, and Hit Rate value. The new rating decreased average of values of RMSE, MSE, and MAE by 62.10%, 53.62%, and 65.97%, respectively, and also increased FCP and Hit Rate by 11.89% and 31.42%.

NCR generally improved accuracy across metrics such as RMSE, MSE, MAE, FCP, and Hit Rate, but Last.fm dataset presented an exception where good MSE and MAE values were achieved without NCR intervention. This anomaly might have risen from dataset's low-density value and the abundance of outliers in the rating distribution, as shown from the boxplot in Fig. 6. Consequently, further research would be needed to validate the observations.

This research had certain limitations, as the first method sacrificed computing performance when handling new rating in a large dataset. The second method required rating to be normalized to a scale of 1–5 before implementing NCR.

Acknowledgements
Not applicable.

Author contributions
Tora Fahrudin proposed the method, prepared the dataset, literature study researched, and wrote the manuscript.Dedy Rahman Wijaya conducted an experiment and analyzed the result. All authors read and approved the final manuscript.

Funding
Telkom University.

Data availability
No datasets were generated or analysed during the current study.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
The authors provided consent for publication.

Competing interests
The authors declare no competing interests.

Received: 31 January 2024 / Accepted: 21 June 2024
Published online: 02 July 2024

References

1. Ben Schafer J, Konstan J, Riedl J. Recommendation systems in E-Commerce, in *Proceedings of the 1st ACM conference*, 1999. [Online]. Available: https://dl.acm.org/doi/10.1145/336992.337035.
2. Sun Z, et al. Recommendation systems based on social networks. J Syst Softw. Jan. 2015;99:109–19. https://doi.org/10.1016/j.jss.2014.09.019.
3. Raza S, Ding C. News recommendation system: a review of recent progress, challenges, and opportunities. Artif Intell Rev. Jan. 2022;55(1):749–800. https://doi.org/10.1007/s10462-021-10043-x.
4. He Z, Hui B, Zhang S, Xiao C, Zhong T, Zhou F. Exploring indirect entity relations for knowledge graph enhanced recommendation system. Expert Syst Appl. Mar. 2023;213. https://doi.org/10.1016/j.eswa.2022.118984.
5. Falk K. *Practical Recommendation Systems*, 1st ed., vol. 1. Shelter Island: Manning, 2019. Accessed: Dec. 02, 2022. [Online]. Available: https://www.amazon.com/Practical-Recommendation-Systems-Kim-Falk/dp/1617292702#detailBullets_feature_div.
6. Tran TNT, Felfernig A, Trattner C, Holzinger A. Recommendation systems in the healthcare domain: state-of-the-art and research issues, *J Intell Inf Syst*, vol. 57, no. 1, pp. 171–201, Aug. 2021, https://doi.org/10.1007/s10844-020-00633-6.
7. Fahrudin T, Wisna N. The exploration of Restaurant Recommendation System. J Comput Sci. 2022;18(8):784–91. https://doi.org/10.3844/jcssp.2022.784.791.
8. Ko H, Lee S, Park Y, Choi A, A Survey of Recommendation Systems: Recommendation Models, Techniques, and, Fields A. *Electronics (Switzerland)*, vol. 11, no. 1. MDPI, Jan. 01, 2022. https://doi.org/10.3390/electronics11010141.
9. Roy D, Dutta M. A systematic review and research perspective on recommendation systems. J Big Data. Dec. 2022;9(1). https://doi.org/10.1186/s40537-022-00592-5.
10. Thiengburanathum P, Cang S, Yu H. A Decision Tree based Recommendation System for Tourists, in 2015 *21st International Conference on Automation and Computing (ICAC)*, 2015. [Online]. Available: www.tripadvisor.com.
11. Pujahari A, Padmanabhan V. An approach to content based recommendation systems using decision list based classification with k-DNF rule set, in *Proceedings – 2014 13th International Conference on Information Technology, ICIT* 2014, Institute of Electrical and Electronics Engineers Inc., Feb. 2014, pp. 260–263. https://doi.org/10.1109/ICIT.2014.13.
12. Xu JA, Araki K. A SVM-based Personal Recommendation System for TV Programs, in 2006 *12th International Multi-Media Modelling Conference*, 2016. [Online]. Available: http://www.ontvjapan.com/program/.
13. Yarahmadi Gharaei N, Dadkhah C, Daryoush L. Content-based Clothing Recommendation System using Deep Neural Network, in *26th International Computer Conference, Computer Society of Iran, CSICC* 2021, Institute of Electrical and Electronics Engineers Inc., Mar. 2021. https://doi.org/10.1109/CSICC52343.2021.9420544.
14. Fayyaz Z, Ebrahimian M, Nawara D, Ibrahim A, Kashef R. Recommendation systems: Algorithms, challenges, metrics, and business opportunities, *Applied Sciences (Switzerland)*, vol. 10, no. 21, pp. 1–20, Nov. 2020, https://doi.org/10.3390/app10217748.

15. Nassar N, Jafar A, Rahhal Y. Multi-criteria collaborative filtering recommendation by fusing deep neural network and matrix factorization. J Big Data. Dec. 2020;7(1). https://doi.org/10.1186/s40537-020-00309-6.
16. Nilashi M, Ibrahim O, Bagherifard K. A recommendation system based on collaborative filtering using ontology and dimensionality reduction techniques. Expert Syst Appl. Feb. 2018;92:507–20. https://doi.org/10.1016/j.eswa.2017.09.058.
17. Do M-PT, Van Nguyen D. Model-based approach for Collaborative Filtering, in *The 6th International Conference on Information Technology for Education (IT@EDU2010)*, 2010. [Online]. Available: https://www.researchgate.net/publication/321753015.
18. Pujahari A, Sisodia DS. Model-Based Collaborative Filtering for Recommendation Systems: An Empirical Survey, in 2020 *First International Conference on Power, Control and Computing Technologies (ICPC2T)*, 2022.
19. Afoudi Y, Lazaar M, Al Achhab M. Hybrid recommendation system combined content-based filtering and collaborative prediction using artificial neural network. Simul Model Pract Theory. Dec. 2021;113. https://doi.org/10.1016/j.simpat.2021.102375.
20. Walek B, Fojtik V. A hybrid recommendation system for recommending relevant movies using an expert system. Expert Syst Appl. Nov. 2020;158. https://doi.org/10.1016/j.eswa.2020.113452.
21. Badriyah T, Ramadhan YG, Syarif I. An Efficient Framework of Hybrid Recommendation System based on Multi Mode, in *IES 2020 - International Electronics Symposium: The Role of Autonomous and Intelligent Systems for Human Life and Comfort*, Institute of Electrical and Electronics Engineers Inc., Sep. 2020, pp. 351–355. https://doi.org/10.1109/IES50839.2020.9231842.
22. Lu J, Wu D, Mao M, Wang W, Zhang G. Recommendation system application developments: a survey. Decis Support Syst. 2015;74:12–32. https://doi.org/10.1016/j.dss.2015.03.008.
23. Hasanzadeh S, Fakhrahmad SM, Taheri M. Review-Based Recommendation Systems: A Proposed Rating Prediction Scheme Using Word Embedding Representation of Reviews, *Computer Journal*, vol. 65, no. 2. Oxford University Press, pp. 345–354, Feb. 01, 2022. https://doi.org/10.1093/comjnl/bxaa044.
24. Liu Y, Yang J. Improving Ranking-based recommendation by social information and negative similarity, in *Procedia Computer Science*, Elsevier B.V., 2015, pp. 732–740. https://doi.org/10.1016/j.procs.2015.07.164.
25. Guan X, Li CT, Guan Y. Matrix Factorization with Rating Completion: An Enhanced SVD Model for Collaborative Filtering Recommendation Systems, *IEEE Access*, vol. 5, pp. 27668–27678, Nov. 2017, https://doi.org/10.1109/ACCESS.2017.2772226.
26. Kowalczyk W, Szlávik Z, Schut MC. The impact of recommendation systems on item-, user-, and rating-diversity, in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2012, pp. 261–287. https://doi.org/10.1007/978-3-642-27609-5_17.
27. Peedika C. How IMDb ratings work: A Scientific Approach, Cinema Peedika.
28. Huurdeman H, Kamps J, Koolen M, Van Wees J. Using Collaborative Filtering in Social Book Search, in *CLEF 2012*, 2012.
29. Khrisnan S, Patel J, Franklin MJ, Goldberg K. A methodology for learning, analyzing, and mitigating social influence bias in recommendation systems, in *RecSys '14: Proceedings of the 8th ACM Conference on Recommendation systems*, 2018, p. 442.
30. Spiliotopoulos D, Margaris D, Vassilakis C. On exploiting rating prediction accuracy features in dense collaborative filtering datasets. Inform (Switzerland). Sep. 2022;13(9). https://doi.org/10.3390/info13090428.
31. Lestari S, Bharata Adji T, Ema Permanasari A. NRF: Normalized Rating Frequency for Collaborative Filtering Paper, in 2018 *International Conference on Applied Information Technology and Innovation (ICAITI)*, 2018, pp. 19–25. [Online]. Available: http://grouplens.org/datasets/.
32. Gojali S, Khodra ML, Aspect based sentiment analysis for review rating prediction, in. 2016 *International Conference on Advanced Informatics: Concepts, Theory and* Application, *ICAICTA 2016*, 2016. https://doi.org/10.1109/ICAICTA.2016.7803110.
33. Sejwal VK, Abulaish M, Jahiruddin. Crecsys: a context-based recommendation system using collaborative filtering and lod. IEEE Access. 2020;8:158432–48. https://doi.org/10.1109/ACCESS.2020.3020005.
34. Boratto L, Carta S, Fenu G. Investigating the role of the rating prediction task in granularity-based group recommendation systems and big data scenarios. Inf Sci (N Y). Feb. 2017;378:424–43. https://doi.org/10.1016/j.ins.2016.07.060.
35. Nichols DM. Implicit Rating and Filtering, 1998.
36. Núñez-Valdez ER, Quintana D, González Crespo R, Isasi P, Herrera-Viedma E. A recommendation system based on implicit feedback for selective dissemination of ebooks, *Inf Sci (N Y)*, vol. 467, pp. 87–98, Oct. 2018, https://doi.org/10.1016/j.ins.2018.07.068.
37. Choi K, Yoo D, Kim G, Suh Y. A hybrid online-product recommendation system: combining implicit rating-based collaborative filtering and sequential pattern analysis. Electron Commer Res Appl. Jul. 2012;11(4):309–17. https://doi.org/10.1016/j.elerap.2012.02.004.
38. Bai X, et al. Recommendation algorithms for Implicit Information. in Service Science, Management, and Engineering: theory and applications. Elsevier; 2012. pp. 77–94. https://doi.org/10.1016/B978-0-12-397037-4.00005-3.
39. Adomavicius G, Huang Z, Tuzhilin A. Personalization and Recommendation Systems, in *State-of-the-Art Decision-Making Tools in the Information-Intensive Age*, INFORMS, 2008, pp. 55–107. https://doi.org/10.1287/educ.1080.0044.
40. Cantador In et al. Personalized Pricing Recommendation System, in *HetRec '11: Proceedings of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommendation Systems*, ACM, 2011, p. 77.
41. Guerreiro J, Rita P. How to predict explicit recommendations in online reviews using text mining and sentiment analysis, *Journal of Hospitality and Tourism Management*, vol. 43, pp. 269–272, Jun. 2020, https://doi.org/10.1016/j.jhtm.2019.07.001.
42. Srifi M, Oussous A, Lahcen AA, Mouline S. Recommendation systems based on collaborative filtering using review texts-A survey, *Information (Switzerland)*, vol. 11, no. 6, Jun. 2020, https://doi.org/10.3390/INFO11060317.

Publisher's Note