Identifying MCCF Vulnerabilities and Preparing Datasets for Effective Analysis: A Comprehensive Approach

Shraddha Soni[1*] and Sunita Varma[2]

[1]I.I.P.S, D.A.V.V.,Indore, India
shraddha.soni@iips.edu.in*

[2] S.G.S.I.T.S, Indore, India
sunita.varma19@gmail.com

Abstract. Recommender system has added an incredible convenience by delivering suggestions that are of our interest while we dive into the ocean of information. Among different recommender techniques, Multi-criteria collaborative filtering (MCCF)system is the one who determines user preferences by carefully considering the user rating on multiple criteria of an item. As the system is dependent on explicit ratings given by user, this openness may make the entire system vulnerable for shilling attacks, thereby causing malicious users to corrupt the credibility of system by proving fake ratings. Identification of all such vulnerabilities within these systems provides worthwhile insight for the security and robustness of this recommender system. This study aims to identify various vulnerabilities of MCCF and construction of multiple datasets for specific analytic requirements. The finding underpins measures that are to be taken for building a robust system and the datasets constructed can be used to analyze and study shilling attacks on MCCF more precisely in future researches.

Keywords: Recommender system, Multi-criteria collaborative filtering (MCCF), vulnerabilities, shilling attacks, collaborative filtering, datasets

1 Introduction

Due to affordable internet plans we have been surrounded enormously by the information all around. Getting the personalized or relevant information for decision making has turned into a challenge. Recommender systems (RS) are the tools that substantially aid in overcoming this problem by providing meaningful recommendation as per the requirements to the active user. To have a deeper insight we begin with the related concepts of recommender system, MCCF used in this study and shilling attacks on these systems in section 1.1 ,1.2 and 1.3. In the present time Recommender system is one of the interesting and involved area of research

1.1 Recommender System

RS are being used by almost every platform be it e-commerce websites, music apps, OTT platforms, online learning platforms, hotel booking or anything. These systems leverage the sources of information like explicit ratings, links clicked, time spent on page, browsing behavior etc. to infer the user interest and suggest the items conse-

S. Bhalerao et al. (eds.), *Proceedings of the 2nd International Conference on Recent Advancement and Modernization in Sustainable Intelligent Technologies & Applications (RAMSITA-2026)*, Advances in Intelligent Systems Research 207,
https://doi.org/10.2991/978-94-6239-678-4_21

quently. We have seen suggestions like-"People you may know"," friend suggestions", "similar products or products related to this item", "styling ideas" etc. while we browse various websites. There are many approaches to recommendation- Collaborative filtering (CF), content based and hybrid approach are the most popular (see **Fig.1**). All having their advantages and disadvantages [1]. Originally, the "Collaborative Filtering" term was introduced in 1992 [2]. CF is the most popular technique in the literature. This method uses ratings given to the items in the past by users with similar preferences, for predicting the significance of an unexplored item for an active user. In Content based filtering the system generates suggested item list by using the features of the previously rated items by the concern user. Hybrid recommendation uses the combination of both or any other techniques [3].

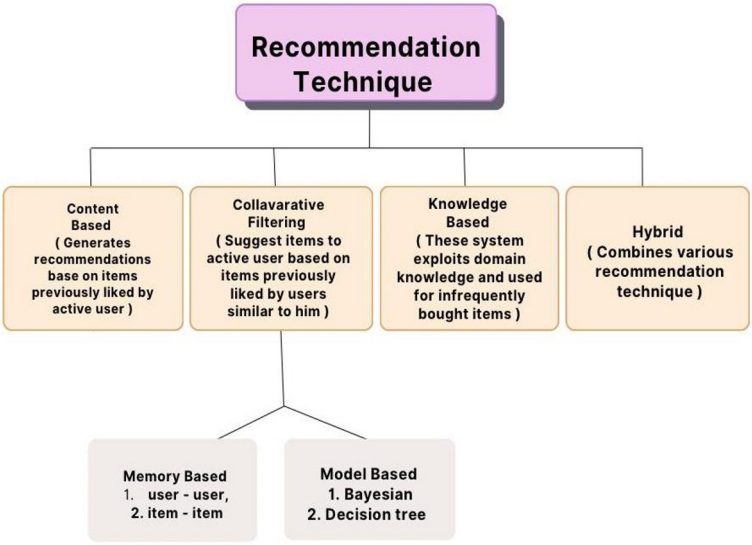

Fig. 1. Recommender System techniques

1.2 Multi-criteria collaborative filtering

As mentioned above, CF is most commonly used techniques, uses single rating given by users on experienced item, is usually overall rating. This rating implicitly covers various parameters for evaluating an item. In user-user CF, to get the prediction score $\hat{r}_{ui}$ for user u on unseen item i ,we calculate similarity of user u with other users in the system using their past rating on common items and taking *top-n* similar users. The rating set of these similar users are used to get the prediction score for unseen item [4]. To measure the similarity among users, various functions like Pearson correlation coefficient, Cosine similarity, Euclidean distance are commonly used. The downside of this single rating is that it does not let out which attribute of an item was considered by user while rating it. For instance, if user *u1* and *u2* both rate a hotel with maximum rating as 5, *u1*and *u2* might have location and room service as their preference respectively

while giving ratings. Despite the same ratings, both users' choices are different and consequently, this hidden liking details may generate irrelevant suggestions.

MCCF extends the traditional CF [5][3]. This enables the user to rate an item on its multiple criteria. An item is usually described by the features called criteria. For example, a movie can have rating criteria as like graphics, music, genre, sound effect etc. MCCF helps in providing the deeper understanding of which criteria are being liked or disliked by an individual and up to what extent. For such user-based methods, it is desirable to correctly determine the most similar users to the target user. Hence, this quality feedback in the form of ratings on multiple criteria significantly improves the accuracy of the system. In traditional single rating CF, rating prediction of an item for a user is made with utility function as in eq (1):

$$F': \text{Users} \times \text{Items} \rightarrow r(o) \qquad \text{eq (1)}$$

Here r(o) is the single overall rating predicted confined to a range for a user item interaction. MC systems can pick overall rating along with all criteria ratings for the utility function or it can use only criteria ratings. Hence, the utility function for multi-criteria system can be formulated in following ways as in eq (2) and eq (3) [6]:

$$F': \text{Users} \times \text{Items} \rightarrow r(o) \times r(1) \times r(2) \ldots . r(n) \qquad \text{eq (2)}$$

$$F': \text{Users} \times \text{Items} \rightarrow r(1) \times r(2) \ldots . r(n) \qquad \text{eq (3)}$$

Table 1 shows the rating dataset of user item interaction for neighborhood based single criteria CF. Considering *User1* as an active user, the predicted rating of *Item5* for *User1* is evaluated by his closest user who rated *Item5*. Choice of *User3* matches with *User1* and predicts the rating as 9 of *Item5* for *User1*.

Table 1. User-Item interaction in single criteria

	Item1	Item2	Item3	Item4	Item5
User1	6	4	6	4	?
User2	10	9	1	9	5
User3	6	4	6	4	9
User4	5	5	5	5	3

Now, considering MCCF for same scenario with items rated on four different criteria in table 2. By observing these multi-criteria ratings, it is evident that *User4* matches more closely with *User1* than *User2* in their preferences. *User3* predicts the rating of *Item5* for *User1* as 3 which greatly contrast with the rating in above case because here the criteria ratings describe the reasons for likes and dislikes and also up to what extent as compared to single rating CF that signifies only the extent of likes or dislikes for items.

Table 2. User-Item interaction in multi-criteria

	Item1	Item2	Item3	Item4	Item5
User1	6(7,7,2,2)	4(1,1,5,5)	6(4,4,1,1)	4(3,3,1,1,)	?
User2	10(9,9,8,8)	9(8,8,8,8)	1(1,1,1,1)	9(7,7,8,8)	5
User3	6(2,2,7,7)	4(5,5,1,1)	6(1,1,4,4)	4(1,1,3,3)	9
User4	5(6,6,3,2)	5(2,2,5,5)	5(5,5,2,2)	5(4,4,2,2)	3

1.3 Shilling attacks and attack profiles

The item rating in CF is very crucial because the quality and authenticity of this data determine the accuracy of the system. However, apart from genuine users some malicious users can also easily register on the concern platform to provide ratings to the items. These malicious users insert shill profiles to attack the system with the intent to harm the system's performance by supplying fake ratings. As per the intent, these attacks can be broadly classified into push and nuke attacks. Push attack promotes the target item by giving the highest rating through malicious users, thereby increasing its recommendation frequency. These users in nuke attack intents to demotes the target item by rating it to lowest rating [7]. Typically, a shilling attack profile has the following structure shown in table 3:

Table 3. Attack profile structure

I(S) (selected items)	I(F) (filler items)	I(ϕ) (not rated items)	I(T) (target items)
$i_1(S)…i_j(S)$	$i_1(F)…i_k(F)$	$i_1(\phi)…i_m(\phi)$	$i_1(T)…i_n(T)$
$\alpha(i_1(S))…\alpha(i_j(S))$	$\beta(i_1(F))…\beta(i_k(F))$	No ratings	$\rho(i_1(T))…\rho(i_n(T))$
Ratings of I(S)	Ratings of I(F)	Not rated	Ratings of I(T)

Above attack profile consists of four sets. Set I(S) have selected items with certain special characteristics required in specific attack strategies. This set is not necessary for some attack models. Filler item I(F) set contains randomly selected items that help the attackers to hide themselves within the system. I(ϕ) set of unrated items in attack profile. Item(s) whose recommendation frequency is to be increase or decrease is in target item set I(T) and rated with either highest or lowest rating score [8]. Based on the attacker's intent and knowledge needed to mount an attack, different attack models have been discussed in [9]. Following are some well-known attack models with their summary in table 9: **Random attack***:* This comes under low knowledge attack. Set I(S) is null for this model. Filler items are randomly chosen and allotted with rating from normal distribution with mean and standard deviation across all items [6]. Target item I(T) rated with highest (lowest) rating for push(nuke) attack. **Average attack:** It is a high knowledge attack as it requires mean rating of each item across the users who have rated it. This attack is similar to random attack except that the filler items are given with their mean ratings [6]. **Bandwagon attack:** Attackers exploits popular items, which are rated by many users, in the I(s) set. Items in this set are rated with maximum rating along with the maximum rating to items in I(t) set. Fillers are selected and rated in the same way as in the random attack. **Reverse bandwagon attack:** This attack has resemblance with bandwagon attack except it is used to nuke an item. It uses items which are infrequently rated by people in it's I(s) set. These items and the target item are put with the lowest rating score. Random ratings are given to all items in I(F). **Segment attack:** Idea of this attack is to boost the target item among the targeted group of users having certain preferences. For I(s), items are identified as those which are preferred by this targeted group and put them with highest rating alongside the target item. Items in I(F) are filled with lowest score. **Love/hate attack:** Attack pro-

files for this attack takes random filler items with highest rating value and target item(s) with minimum. This attack can be performed to push an item by swapping the ratings of filler items and target item.

Our work intent to identify vulnerabilities in MCCF and prepares different datasets that can help in dealing with these vulnerabilities. This paper is organized into following key sections. An overview of recent advancement in MCCF is discussed in Section 2. Identified vulnerabilities are mentioned in section 3. Section 4 presents steps to create datasets. Paper is concluded with future scope in section 5.

2 Related work

By evaluating user's choice across various criteria, MCCF can discover their priorities and requirements more accurately, facilitating for suggesting relevant items. Presently, many of the researches are focusing on challenges such as cold start problem, dealing with sparsity, improving accuracy, scalability etc. in addition to these, another obstacle is, it is uncommon to find large and good multi-criteria datasets for algorithm evaluation [10]. As these systems heavily rely on user to provide rating data for various criteria, there arises a consequential problem of sparsity. Since, conventionally, only a very small proportion of items is rated by users, sparsity is a problem common to almost every recommender system [6]. Many different algorithms are presented to address this issue. An ESAE [11] approach transforms sparse rating into dense matrix by integrating multi-criteria rating of an item and utilizing them to predict overall rating. In another approach [12], Ishwari et al. integrated deep learning with SVD and auto encoders to predict unknown ratings for populating the rating gaps. Recently, a novel NLP-based BERT model [13] addressed the sparsity by analyzing reviews or textual information to generate numeric data, and this dense data provides remarkable accuracy in prediction. Embedding trust value in multi-criteria system is another possibility to manage sparsity [14]. Since MCCF aims to deduce users' preferences using these explicit criteria ratings, accuracy becomes a key concern. Matrix factorization and HOSVD based algorithm proposed by Bokde et al. in [15] provides quality recommendations with accuracy. Nilashi et al. in [16] enhance accuracy by Expectation maximization and PCA. In his one more work [17] Nilashi applied supervised and unsupervised algorithm with ratings and reviews to predict relevant product even with sparse dataset. An AGA based model [18] for Multi-criteria RS is more effective than single rating technique. An auto-encoder based algorithm [19] improved accuracy by capturing complex user-item interaction for getting user preferences and the algorithm performs notably better than other state of art algorithms. Besides sparsity, Ishwari et al. also addressed accuracy in [20] using AM-MCRS methodology. They used adaptive attention to identify criteria preferred by users with past interactions and dynamically weight them for generating more accurate suggestions by applying nearest neighbor. Although these systems are more detailed and accurate compared to traditional single rating systems, problem with many explicit ratings is the intrusiveness where intruders can easily influence the system's suggestions by forming malicious profiles. The intruder could be either fake users or producer itself. The purpose is to create malicious profiles that rate target items with other items in such a manner that they resemble regular user's profile, thereby manipulating actual rating data for generating inappro-

priate suggestions. This act is called shilling attack. Many studies [21][22][23] exhibit that CF technique is susceptible to attacks. As an extension of collaborative filtering, MCCF also suffers the same. Since, MC systems use item ratings based on various criteria provided by the nearest users to predict ratings for unseen items, they are highly vulnerable to such attacks. MC system also exploits implicit feedback by observing user-system interaction like time spent on page, browsing, frequency of accessing an item, clicking to determine user preference. Various studies integrate explicit ratings with other system data to enhance the accuracy and deal intrusiveness. Though implicit feedback leverages the value of users' indirect participation to reduce intrusiveness, it cannot predict user's choice precisely [24] and also implicit feedback could be expensive with respect to computation and storage. However, a hybrid approach proposed in [25] uses MC explicit ratings and user's text reviews as being implicit data to combine with feature space to reach highly accurate predictions. Furthermore, researches on shilling attacks can be split into three distinct categories as shill profiles detection, designing robust algorithm and analyzing robustness of algorithms against these attacks [5]. Due to explicit ratings, MC systems are open in nature, making them highly vulnerable towards shilling attacks. As malicious profiles are very similar to many authentic ones, it is difficult to identify them. This close similarity poses a big challenge, as the removal of these profiles may recklessly eliminate real profile as well. Various dimensions of MCCF such as accuracy, sparsity, scalability have received significant attention, while shilling attacks are still to be explored in depth. There are limited researches addressing shilling attacks on MCCF. A clustering based robust MCCF algorithm against shilling attack presented in [5]. They categorized the users according to their actual preference and employed DU metric of each group to exclude suspicious ratings. In their further work [26], they analyzed MCCF systems against shilling attacks. For this, they extended popular shilling attack types for CF to influence MCCF algorithms. They also presented a novel attack scheme, named as mode attack. Their experimental results exhibit vulnerability of MC systems to these attacks. Recently, four variants of a novel attack- referred to as Power Item Attack were introduced in [27]. Here, a classification-based shilling profile detection method is also proposed that utilizes generic and user-specific attributes including new features derived from item popularity and rating distribution value. As opposed to single criteria traditional collaborative filtering, it is harder to design, detect and identify shill profiles for MCCF. As stated above, this technique immensely depends on user input to create the suggested item list. The input, in the form of item ratings provides specific details on each criterion to remarkably enhance personalization; however, such specific details given by user may also subject him to privacy risk. A study [28] examines current privacy risks and highlights the need for the development of mechanism to safeguard privacy in MC systems. Besides, they presented [29] privacy preserving entropy-based procedures with minimum sacrifice on accuracy. While many issues of MC systems are discussed in literature, significant efforts are yet required to balance performance, accuracy and system security against shilling attack. Currently, MCCF is one of the growing techniques that require to be strengthening across various aspects and eliminate vulnerabilities to assure a robust system.

3 Identified vulnerabilities in MCCF

One of the purposes of this study is to identify the key factors that make the system vulnerable, thereby increasing the possibility of shilling attacks. Through the detailed literature review, we observed that certain risk factors are directly or indirectly associated with the MC system. Understanding and addressing these factors is vital for developing an accurate and robust system. Overlooking these vulnerabilities may expose the system to threats and negatively impact user experience and trust.

3.1 Limited accessibility of quality Multi-criteria dataset:

For any system a well-designed algorithm and high-quality datasets for testing play a crucial role. Similarly, to effectively assess a robust MCCF algorithm, there is a need for diverse, good-quality and large-scale MC datasets — which, unfortunately, are not easily available. Due to the difficulty in obtaining the datasets, algorithms are often not thoroughly evaluated from different perspectives. Consequently, they become vulnerable and victim of shilling attacks in real-world deployment.

3.2 Widespread adoption of explicit ratings:

While MC ratings can be expressed explicitly, implicitly or together, explicit ratings are more popular. As the shill profiles are designed using explicit ratings, this makes it easier for an attacker to fabricate malicious or shill profiles with these explicit ratings to manipulate the system and this is more straightforward in contrast to implicit ratings.

3.3 Attacks without rating database:

Many attack types require prior knowledge of rating database, while others can be performed with least information- in certain attacks, merely item popularity is enough to mount the attack successfully.

3.4 Limited and incorrect user ratings:

Rating a greater number of criteria can also be exhausting or make users feel insecure about their privacy. Consequently, users either skip rating or submit incorrect ratings. In the former case, the system suffers from cold start problem [28], allowing attackers to make room for their product. In latter case, attackers can easily hide among incorrect ratings and it becomes harder to discriminate between fake and real users. Hence, a malicious user can make the system vulnerable for attack.

3.5 Privacy threat:

In MCCF users rate items across multiple criteria, to strengthen the quality of recommendations. However, based on their ratings and preferences, it's also possible to infer personal traits such as lifestyle, financial status, and demographic details [28]. A malicious actor can infringe user privacy by inferring sensitive information from these data. Without strong privacy measures, MCCF filtering systems are vulnerable to privacy threats.

4 Dataset preparation

4.1 Problem addressed

The existing literature on shilling attacks for MC domain is quite limited. To the best of our knowledge, only a few papers discuss shilling attacks with multi-criteria. It is also difficult to find datasets of MC rating, and even when found, they are often very large in size due to large number of ratings on various criteria. This huge data raises space complexity and needs to be reduced for experiments. Since user doesn't always give ratings to all criteria due to their different rating behavior, we have diversity in rating pattern of the dataset. Also, due to the lack of direct availability of datasets with such diverse rating, we cannot analyze the same MC algorithm across different rating datasets. Furthermore, if we look at the vulnerabilities mentioned above, one of them suggests that even item's popularity solely can be used to mount certain shilling attacks. If we construct different datasets based on these diverse ratings, we can test more precisely the robustness of our algorithms against shilling attacks. And, availability of a such dataset would be needed in future researches for the study, performance evaluation and development of robust MCCF algorithms.

4.2 Dataset description

In this study, we employed raw Yahoo movie multi-criteria data shared by the author of [26] on individual request. The related data is in structured form and includes movie id, criteria ratings, and overall ratings in different files. Some movies have overall rating but no criteria rating, and others have criteria ratings but no overall. Some have both, while others have neither. All the ratings are on a scale of 1 to 13, where 13 represents highly appreciated movie and 1 for the least one. The statistics of data is given in table 4.

Table 4. Statistics of data files

	File1	File 2	File 3
#ratings	248486	754938	NA
#user	127811	127828	NA
#movies	8272	8272	8272
Rating scale	[1-13]	[1-13]	NA
#rating parameters	1	4	NA

4.3 Proposed method

1. Having collected the available yahoo movie data, from individual file, we checked for the repeated rows and rating values to be in the range 1-13. If found so, removed the invalid rows from all files.
2. We merged the data from all files on the basis of common columns into a csv file.
3. Integrated dataset is finalized with total rows as 994156 and 5 columns with unique User_id and Movie_id as 127828 and 8272 respectively. The data obtained looks like as shown in table 5.

Table 5. Dataset after merging files

User_id	Movie_id	Category	Criteria rating	Overall rating
1	1808404878	Story	8	12
	1808404878	Acting	6	12
1	1808404878	Directing	9	12
1	1808404878	Visuals	11	12
2	1800198384	Story	13	13
2	1800198384	Acting	13	13
2	1800198384	Directing	12	13

4. Due to the repetition of User_id, Movie_id and overall rating for each User_id -Movie_id pair, the dataset is still large. We converted above dataset into following format (see table 6). Statistics of this consolidate dataset is given in table 7.

Table 6. Dataset in changed format

User_id	Movie_id	Story	Acting	Directing	Visuals	Overall rating
1	180840..	8	6	9	11	12
2	180019..	13	13	12	.....	.....

5. Removed those rows with overall rating column as empty or all columns are empty.

Table 7. Statistics of consolidated dataset

Total rows	Unique users	Unique movies	Rating scale	Rating parameters
248539	127828	8272	[1-13]	5

6. Checked and removed duplicate rows, if any. The filtered dataset now has 248485 rows with unique User_id and Movie_id as 127811 and 8272 respectively. This dataset contains mixed rows. For some User id-Movie_id combinations, ratings are present across all columns. For others, only the 'overall' rating is present, and in some cases, the 'overall' rating is led by ratings in some criteria only.
7. Above dataset is further divided into three subsets as dataset with all ratings (DSAR), dataset with overall rating (DSOR) and dataset with overall and incomplete ratings in remaining criteria (DSOAIC).
8. All three dataset has some popularity bias also (see fig 2,3,4). The details of our three datasets are in table 8.

Table 8. Prepared Dataset

Datasets	Total ratings	Unique users	Unique movies	Rated criteria	Popularity bias
DSAR	927240	94937	7794	All+overall	Yes
DSOR	57929	5906	36435	Only overall	Yes
DSOAIC	18242	3896	2211	Overall+any2	Yes

To cope up with these vulnerabilities of MC Recommender system, it is vital to develop robust algorithms – and it is important to have all kind of data sets as well to

test those algorithms. The steps required to prepare such datasets with diversity in rating pattern are collection, cleaning, merging, reduction and creation. Therefore, every step outlined should be applied, as necessary, to generate varied data sets for feeding into the algorithms for analysis.

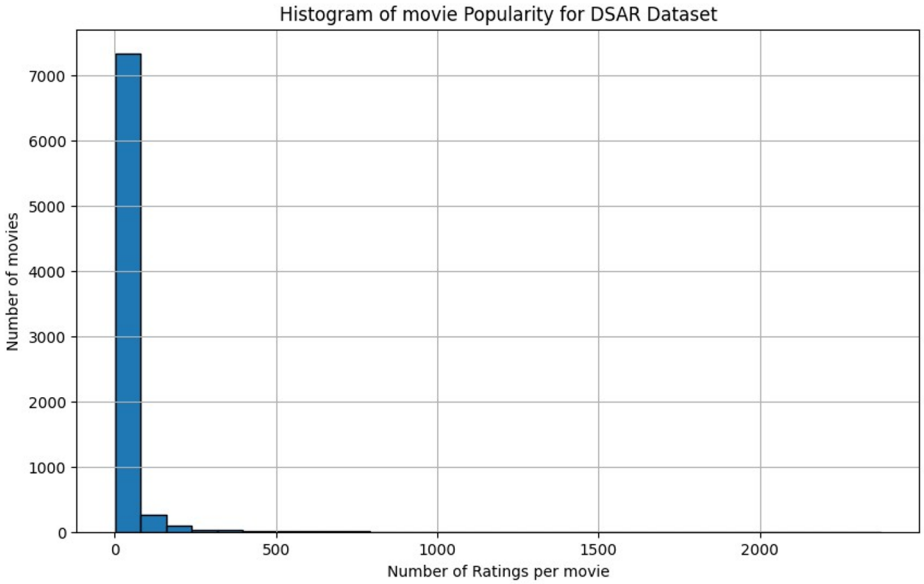

Fig.2(a) Popularity bias and skewness in DSAR

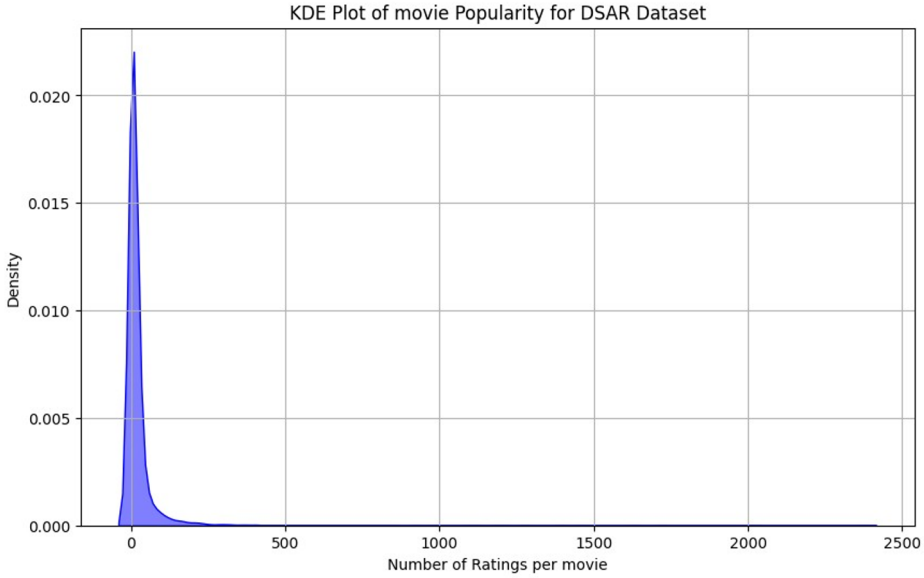

Fig.2(b) Popularity bias and skewness in DSAR

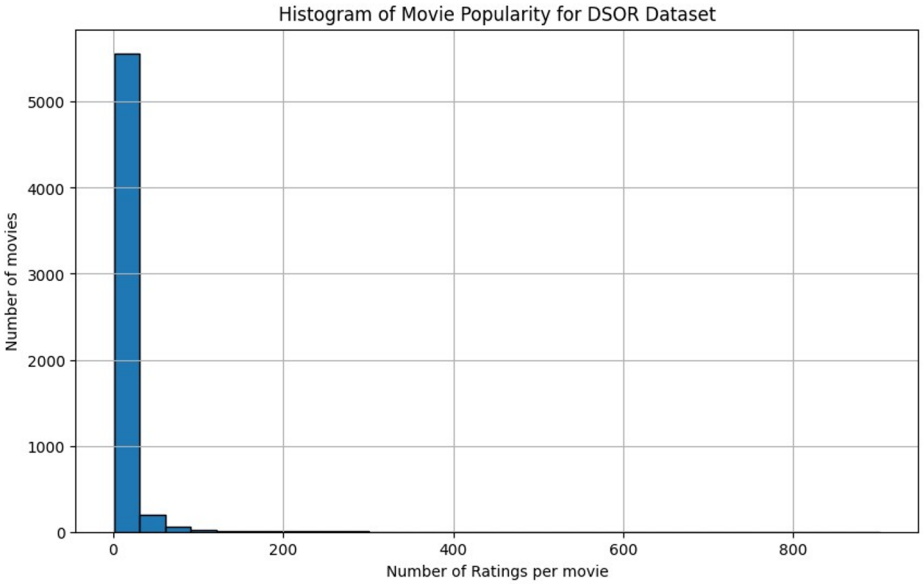

Fig.3(a) Popularity bias and skewness in DSOR

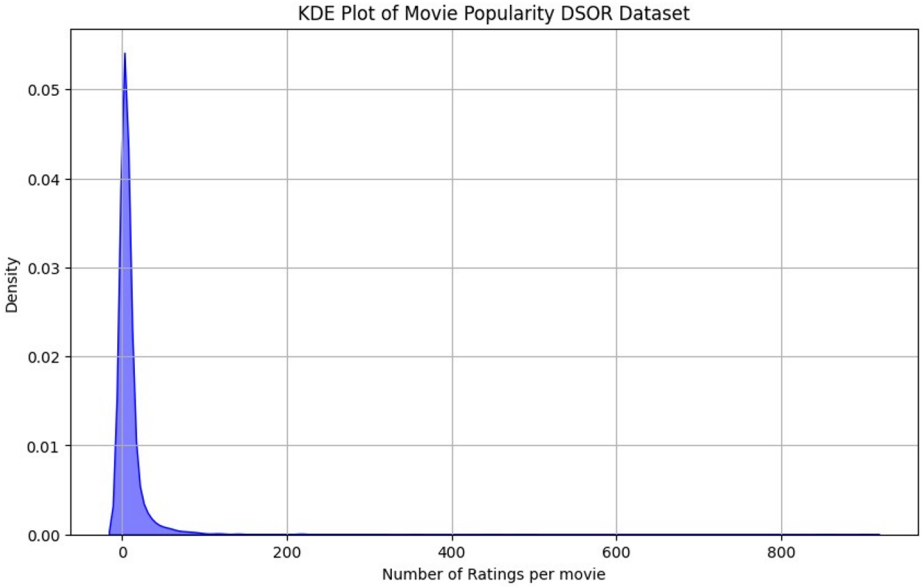

Fig.3(b) Popularity bias and skewness in DSOR

Table 9. Summary of attack types

		Attack models					
		Random	Average	Bandwagon	Reverse Bandwagon	Segment	Love/Hate
I(S)	Selection Method	-	-	Popular	Unpopular	Items preferred by target audience	-
	Rating	-	-	High	Low	High	-
I(F)	Rating	System mean	Item mean	System mean	System mean	Low	High
I(T)	Rating	Low/High	Low/High	High	Low	High	Low
Attack intent		Nuke/Push	Nuke/Push	Push	Nuke	Push	Nuke
Knowledge		Low	High	Low	Low	Moderate	Low

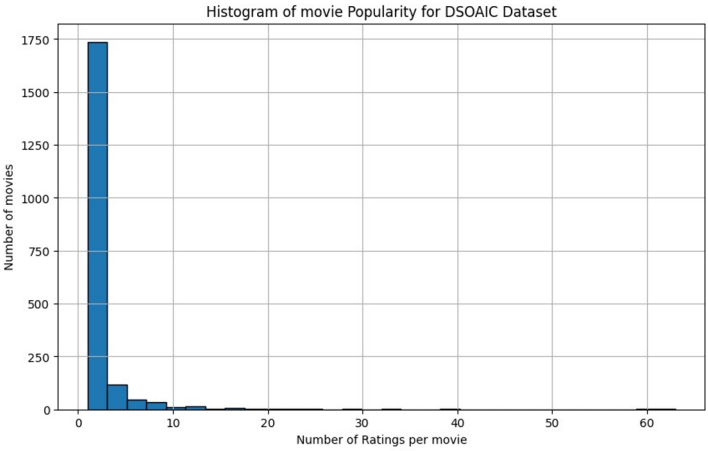

Fig.4(a) Popularity bias and skewness in DSOAIC

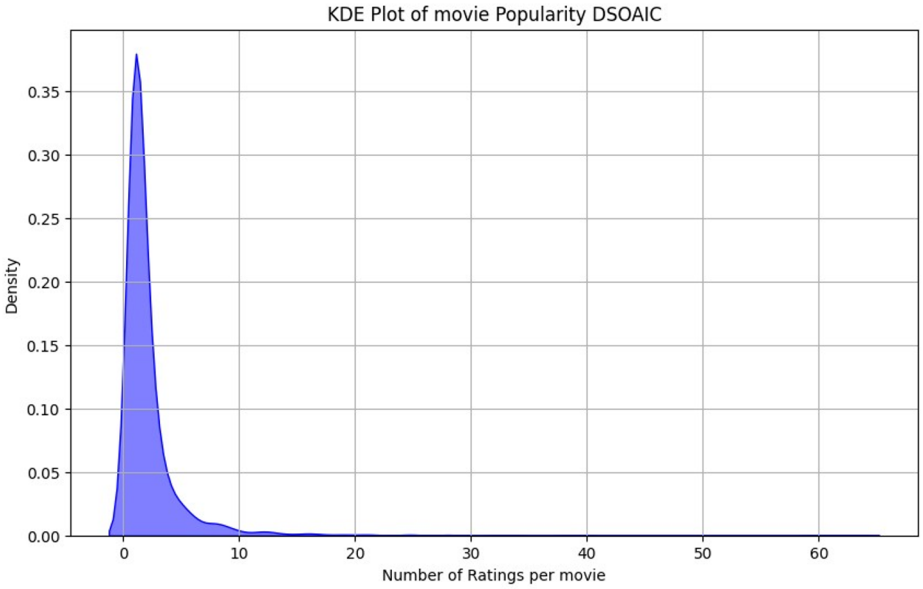

Fig.4(b) Popularity bias and skewness in DSOAIC

5 Conclusion and future work

Despite MCCF's growing relevance to reflect the rationale behind user preferences precisely, these schemes might be prone to be affected by malicious users through various attacks. There are few researches for defense mechanism, malicious profiles identification and development of robust MCCF algorithm against different shilling attacks. In our study, we discussed various vulnerabilities within the system that elevates the possibilities of mounting hostile attacks. These vulnerabilities, in turn, form the basis for preventing MCCF from becoming to be a robust technique. Therefore, investigating its vulnerabilities is worth studying. We also attempted to create basic diverse pattern rating datasets capable of standing MCCF algorithms performance for analysis, evaluation and robustness on such varied data subsets. Many studies for shilling attacks with collaborative filtering occur in the literature, there is very limited research targeting MCCF with shilling attacks. Hence, to deal with the aforementioned vulnerabilities, we need to have better algorithms and their better testing which requires separate datasets with different rating pattern at our disposal. In the future, we intend to explore multi-criteria collaborative filtering behavior across these datasets. Development of these datasets with inherent popularity bias may open new directions in deep understanding of shilling attacks and will certainly be an important step towards development of effective robust algorithms.

References

1. K. Shah, A. Salunke, S. Dongare and K. Antala, "Recommender systems: An overview of different approaches to recommendations", in Proc. 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), Coimbatore, India, 2017, pp. 1-4, doi: 10.1109/ICIIECS.2017.8276172.
2. D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, "Using collaborative filtering to weave an information tapestry," in Journal of Communications of the ACM, vol. 35, no. 12, pp. 61–70, 1992, doi: 10.1145/138859.138867.
3. M. Nilashi, D. Jannach, O. Ibrahim, and N. Ithnin, "Clustering- and regression-based multi-criteria collaborative filtering with incremental updates," in Journal of Information Sciences,vol.293,no.6,pp.235–250,2015,doi: 10.1016/j.ins.2014.09.012.
4. R. Zhang, Q. Liu, C. Gui, J.-X. Wei, and H. Ma, "Collaborative filtering for recommender systems," in Proc. 2nd Int. Conf. Advanced Cloud and Big Data (CBD), 2014, pp. 301–308, doi: 10.1109/CBD.2014.47.
5. A. M. Türk and A. Bilge, "A Robust Multi-Criteria Collaborative Filtering Algorithm," in Proc. 2018 Innovations in Intelligent Systems and Applications (INISTA), 2018, doi: 10.1109/INISTA.2018.8466289.
6. F. Ricci, L. Rokach, and B. Shapira, "Introduction to Recommender Systems Handbook," in Recommender Systems Handbook, Eds. Springer, 2011, pp. 1–35.
7. Z. Yang and Z. Cai, "Detecting abnormal profiles in collaborative filtering recommender systems," in Journal of Intelligent Information Systems, vol. 48, no. 3, pp. 499–518, 2017, doi.org/10.1007/s10844-016-0424-5
8. K. Chen, P. P. K. Chan, F. Zhang, and Q. Li, "Shilling attack based on item popularity and rated item correlation against collaborative filtering," in Journal of International Journal of Machine Learning and Cybernetics, vol. 10, no. C, 2019, doi: 10.1007/s13042-018-0861-2.
9. I.Gunes, C. Kaleli, A. Bilge, and H. Polat, "Shilling attacks against recommender systems: a comprehensive survey," in Journal of Artificial Intelligence Review, vol. 42, pp. 767–799, 2014, doi: 10.1007/s10462-012-9364-9.
10. G. Adomavicius and Y. Kwon, "Multi-criteria recommender systems," in Recommender Systems Handbook, 2nd ed., Eds. Springer, 2015, pp. 847–880.
11. A. Shanmugasundaram and K. Uttej Kumar, "Enhancing multicriteria-based recommendations by alleviating scalability and sparsity issues using collaborative denoising autoencoder," in Journal of Computers, Materials & Continua, vol. 78, no. 2, pp. 2269–2286, 2024, doi: 10.32604/cmc.2024.047167
12. I. S. Rajput, A. S. Tewari, and A. K. Tiwari, "An autoencoder-based deep learning model for solving the sparsity issues of multi-criteria recommender system," in Journal of Procedia Computer Science,vol. 235, no. 1, pp.414–425,2024. doi; 10.1016/j.procs.2024.04.041
13. G. Göksel, A. Aydın, Z. Batmaz, and C. Kaleli, "A novel missing value imputation for multi-criteria recommender systems," in Journal of Information Sciences, vol. 712, 2025, Art. no. 122139, doi: 10.1016/j.ins.2025.122139.
14. A. Goswami, Á. P. Dwivedi, and V. Kant, "Trust-enhanced multi-criteria recommender system" in Soft Computing: Theories and Applications, M. Pant et al., Eds., Advances in Intelligent Systems and Computing, vol. 583. Springer, 2018, doi: 10.1007/978-981-10-5687-1_39.
15. D. Bokde, S. Girase, and D. Mukhopadhyay, "An approach to a university recommendation by multi-criteria collaborative filtering and dimensionality reduction techniques," in Proc. 2015 IEEE Int. Symp. Nanoelectronic and Information Systems (iNIS), 2015, pp. 231–236, doi: 10.1109/iNIS.2015.36.

16. M. Nilashi, O. Ibrahim, N. Ithnin, and N. H. Sarmin, “A multi-criteria collaborative filtering recommender system for the tourism domain using Expectation Maximization (EM) and PCA-ANFIS,” in Journal of Electronic Commerce Research and Applications, vol. 14, pp. 542–562, 2015, doi: 10.1016/j.elerap.2015.08.004.
17. M. Nilashi, R. A. Abumalloh, S. Samad, B. Minaei-Bidgoli, H. H. Thi, O. A. Alghamdi, M. Y. Ismail, and H. Ahmadi, “The impact of multi-criteria ratings in social networking sites on the performance of online recommendation agents,” in Journal of Telematics and Informatics, vol. 76, Art. no. 101919, 2023, doi: 10.1016/j.tele.2022.101919.
18. M. Hassan and M. Hamada, "Improving prediction accuracy of multi-criteria recommender systems using adaptive genetic algorithms," 2017 Intelligent Systems Conference(IntelliSys),London,UK,2017,pp.326-330,doi: 10.1109/IntelliSys.2017.8324313.
19. Q. Y. Shambour, “A deep learning based algorithm for multi-criteria recommender systems,”in Journal of Knowledge-Based Systems,vol. 211, Art. no.106545,2021, doi;10.1016/j.knosys.2020.106545
20. I. S. Rajput, A. S. Tewari, and A. K. Tiwari, “Adaptive attention mechanisms based multicriteria recommender systems for improved user preference discovery,” in Journal of SN Computer Science, vol. 6, no. 4, Art. no. 389,2025, doi:10.1007/s42979-025-03919-0.
21. H. Xia, B. Fang, M. Gao, and H. Ma, “A novel item anomaly detection approach against shilling attacks in collaborative recommendation systems using the dynamic time interval segmentation technique,” in Journal of Information Sciences, vol. 306, no. C, pp. 150–165, 2015, doi: 10.1016/j.ins.2015.02.019.
22. K. Patel, A. Thakkar, C. Shah, and K. Makvana, “A state of art survey on shilling attack in collaborative filtering based recommendation system,” in Proc. First Int. Conf. Information and Communication Technology for Intelligent Systems, Vol. 1, Smart Innovation, Systems and Technologies, vol. 50, pp. 377–385, 2016, doi: 10.1007/978-3-319-30933-0_38.
23. F. Rezaimehr and C. Dadkhah, “T&TRS: robust collaborative filtering recommender systems against attacks,” in Journal of Multimedia Tools and Applications, vol. 83, no. 11, pp. 1–31, 2023, doi: 10.1007/s11042-023-16641-x.
24. P. Kathiresan and S. Ramakrishnan, “A study on implicit feedback in multicriteria e-commerce recommender system,” in Journal of Electronic Commerce Research, vol. 11, no. 2, pp. 140–156, 2010.
25. A. Ebadi and A. Krzyżak, “A hybrid multi-criteria hotel recommender system using explicit and implicit feedbacks,” in Journal of International Journal of Computer and Information Engineering, vol. 10, pp. 1450–1458, 2016.
26. A. M. Türk and A. Bilge, “Robustness analysis of multi-criteria collaborative filtering algorithms against shilling attacks,” in Journal of Expert Systems with Applications, vol. 115, pp. 386–402, 2019, doi: 10.1016/j.eswa.2018.08.001.
27. T. Turkoglu, E. Yalcin, and C. Kaleli, “A novel classification-based shilling attack detection approach for multi-criteria recommender systems,” in Journal of Computational Intelligence, vol. 39, no. 3, pp. 499–528, 2023, doi: 10.1111/coin.12579.
28. A. Yargic and A. Bilge, “Privacy risks for multi-criteria collaborative filtering systems,” in Proc. 2017 26th Int. Conf. Computer Communication and Networks (ICCCN), 2017, pp. 1–6, doi: 10.1109/ICCCN.2017.8038509.
29. A. Yargic and A. Bilge, “Privacy-preserving multi-criteria collaborative filtering,” in Journal of Information Processing & Management, vol. 56, no. 3, pp. 994–1009,2019, doi: 10.1016/j.ipm.2019.02.009.