ENHANCING RECOMMENDER SYSTEM SECURITY: DETECTING INFORMED ATTACKS

Jyoti*

Open Scholar, Department of Mathematics
Jind, Haryana, India

***Email ID**: jyotimorptr@gmail.com*

Accepted: 07.06.2023 **Published**: 01.07.2023

Keywords: RMED Attacks, Unsupervised Classification, Clustering Models, Recommender System.

Abstract

Several different fields that are relevant to customized services have started using recommender systems. One of the types of it that is used the most often is called a collaborative filtering-based recommender system. However, there is a problem to it, and that is the fact that it is very susceptible to assaults that include profile injection. As a direct consequence of this, the algorithm will now offer biased forecasts. An attacker has a high degree of ease in manipulating the results of these recommender systems due to the poor performance of these systems against these assaults. A number of investigations have been carried out in this field in order to identify these assaults and lessen the damage that they do. Within the scope of this work, we have discussed the informed assaults, which fall under the category of profile injection attacks as well. In order to differentiate these assault profiles from actual users, a variety of distinguishing characteristics have been uncovered. In addition to this, we discussed unsupervised methods for detecting these kinds of assaults with the use of characteristics.

Paper Identification

**Corresponding Author*

1. Introduction

The integration of a recommendation system has become a fundamental component in the vast majority of e-commerce platforms, including but not limited to Flipkart, Amazon, Edx, You Tube, and Netflix. Recommender systems encompass a diverse range of types, such as collaborative filtering, content-based, hybrid recommenders, and more. Collaborative filtering stands as one of the prevailing and widely employed types of recommendation systems. In the event that two users have exhibited similar preferences in the past, it is plausible to infer that they may also possess similar preferences in subsequent instances. The fundamental principle upon which the system is established is as follows. The platform exhibits a notable drawback referred to as the cold start issue. This issue arises when a user accesses the platform for the first time without any prior association with other user profiles. Consequently, the system encounters challenges in recommending products to the new user. The category of collaborative filtering-based recommender systems encompasses both user-based and item-based recommender systems. One such system is referred to as a user-based recommender system. In a user-based recommender system, a correlation is computed among the users, whereas in an item-based recommender system, the correlation is calculated among the recommended products. The item-based recommender system offers a significant reduction in temporal complexity compared to the user-based recommender system. This is the primary reason why Amazon has chosen to implement it. The prevailing consensus in the field acknowledges that the quantity of users within a given system typically surpasses the quantity of objects. Consequently, this assumption is commonly adopted. Consequently, the duration required to ascertain the connection between the various elements will be reduced.

Recommender systems that utilize collaborative filtering also encounter an additional notable issue. Profile injection attacks can frequently and effortlessly achieve success when targeting the system. During a profile injection attack, the perpetrator will introduce counterfeit user profiles into the system with the intention of manipulating the rating prediction for a particular item or a group of products. The counterfeit user profiles have been meticulously crafted to exhibit an indistinguishable resemblance to authentic user profiles. There are two distinct types of assaults: push attacks and nuke attacks. A push attack involves granting an advantage to a specific item, whereas a nuke attack involves imposing a disadvantage on a particular item. The impact of the attack profile's magnitude is a crucial factor in determining the forecast's outcome. The quantity of fabricated user profiles generated and subsequently uploaded into the system by the attacker is referred to as the "fictitious user profile count." Typically, the perpetrator will opt for automation in executing this process due to the inherent challenge of manually injecting a substantial quantity of attack profiles into a given system. Prior to granting access to the website, it is common for the site administrator to increase the expense associated with profile creation by mandating registration or the utilization of a captcha. The purpose of this measure is to deter the creation of fraudulent profiles. The assessment of the attack profile's effectiveness can be further enhanced by considering the dimensions of the assault profile. The attack profile is defined by the quantity of ratings it assigns. Typically, a genuine user will lack the capacity to furnish ratings for a multitude of items, whereas an automated attack profile will possess the ability to generate ratings. In contrast to an alternative form of attack, a specific type of attack necessitates a significantly higher level of expertise within the pertinent domain. Implementing attacks that require a higher level of expertise is widely recognized as a challenging endeavor. To provide an illustrative example, random assaults and bandwagon attacks can be categorized as instances of low knowledge attacks, whereas segment attacks can be classified as instances of high knowledge attacks. Moreover, content-based recommender systems and hybrid recommender systems are widely employed on commercial platforms. The content-based recommender system operates by providing recommendations to users, leveraging the keywords or descriptions associated with the products that align with their interests. Hacking is a potential method that can be employed to launch an attack on the system. However, it is worth noting that the system is not highly vulnerable to assaults utilizing profile injection. This is due to the fact that only the operator has the authority to input the item's description and keywords on the website. Due to the absence of authorization granted to the third party, any modifications to the information contained within the product description are prohibited.

The rating prediction process in a collaborative filtering-based recommender system occurs in two distinct stages. The initial step in the process entails employing Pearson correlation as the primary tool to ascertain the level of similarity between the user who will be the focal point of the forecast and the other users within the system. The solution can be obtained by utilizing equation (1).

$$S_{u,v} = \frac{\sum_{i\in I}(r_{u,i}-\overline{r_u})(r_{v,i}-\overline{r_v})}{\sqrt{\sum_{i\in I}(r_{u,i}-\overline{r_u})^2}\sqrt{\sum_{i\in I}(r_{v,i}-\overline{r_v})^2}} \qquad (1)$$

Here, $S_{u,v}$ is the similarity between user *u* and user *v*. $r_{u,i}$ is the rating given by the user *u* to item *i*. $\overline{r_u}$ is the average of all the ratings given by the user *u*. *i* is the subset of items *I* rated by user *u* and *v*. This equation gives value in the ranging between -1 to 1. Higher the value of this equation means higher is the correlation and vice versa i.e. if equation 1 generates value 1 between user *u* and *v*, it signifies that both the users *u* and *v* have same or almost same rating pattern in the system. Similarly, if this equation generates value -1 between user *u* and *v*, it signifies that both the users *u* and *v* do not share similar rating pattern. Both BellCore and LensGroup used the pearson correlation to derive the similarity in their research project. In the second step, prediction is calculated as per the equation 2. Top *k* nearest neighbors finds out by using equation 1 are used to calculate the rating prediction.

$$P_{u,i} = \frac{\sum_{n\in Neighbors}(r_{n,i}-\overline{r_n})S_{u,n}}{\sum_{n\in Neighbors}|S_{u,n}|} + \overline{r_n} \qquad (2)$$

$P_{u,i}$ is the predicted rating for the user *u* of item *i*.

2. Informed Attack Models

An attacker creates fake user profiles and awards those profiles with phony ratings inside the system. This causes the system to provide inaccurate recommendations to the real users, which, in turn, causes the real users to make poor choices. These bogus profiles have an adverse effect on a significant number of real users. The sort of attack that takes the least amount of system knowledge is the easiest to plant in the system, and vice versa; but, in general, an assault that requires a low amount of knowledge causes less of a dent in the recommendations made by the system. High-knowledge assaults, which are referred to as informed attack models, are explored in this study. The majority of informed attack models may be broken down into two categories: the probing attack, on the one hand, and the power user assault, on the other.

2.1 Probe Attack

During the course of this attack, the perpetrator generates fictitious user profiles. These profiles are then utilized to assign ratings to a set of randomly chosen items, referred to as seed items. The seed items in the system are assigned ratings, which represent the average rating of each item. The attack profiles additionally provide ratings for the targeted entity. The rating provided is at its highest level for a push attack and at its lowest level for a nuke attack. Recommendations for the target item are generated by analyzing the correlation between genuine user profiles and these fake user profiles. The probe attack methodology offers the attacker a means to systematically gather information regarding the distribution of ratings within the system.

This strategy offers a distinct advantage over the power user attack by requiring less domain knowledge. In contrast to the power user attack, which necessitates the selection of a large number of seed items, this strategy only requires the random selection of a small set of seed items. Subsequently, the recommendation system will then select additional items and generate ratings for them.

2.2 Power User Attack

Power users are defined as users who exhibit the highest degree of correlation with other users within the system, resulting in the maximum number of neighboring users. Typically, these users have assigned ratings to a substantial quantity of items within the system. This practice ensures that there is a shared basis for correlation calculations with other users in the system. The recommendations to other users in the system are also influenced by correlated users. The attacker designates a group of power users within the system as the attack profiles. The quantity of users utilizing this set is contingent upon the magnitude of the assault. The system assigns the highest attainable rating to the designated item or group of items during a push attack scenario. In the event of a nuclear attack, the system assigns the lowest possible rating to the target item or group of target items.

3. Experimental Evaluation

To assess the effectiveness of an attack, the prediction shift of the targeted item within the system is measured. In order to counteract the impact of the attack, these attack profiles are identified and their predictions are disregarded when generating recommendations. Over the past fifteen years, numerous studies have been conducted with the aim of identifying and detecting these attack profiles. Both supervised and unsupervised learning models are employed for the purpose of detecting anomalies. In order to utilize supervised models, it is necessary to label the sample data and subsequently train the classifiers. This tool is employed in situations where the specific type of attack is known. Unsupervised models are employed to detect and classify unknown forms of attacks. In a general context, it can be observed that supervised models tend to exhibit higher levels of accuracy compared to unsupervised models.

3.1 Dataset

In this research, Movielens dataset is used. Description of the dataset is given below in detail:

Table 1: Description of MovieLens dataset.

Attribute	Value
Number of ratings	100836
Number of movies	9724
Maximum user ID	610
Minimum user ID	1
Number of users	610
Least movie ID	1

Maximum movie ID	193609
Maximum number of ratings given by any user	2698
Least number of ratings given by any user	20
Movie ID with maximum 5 ratings	318
Average number of ratings by each user	165.305
Average number of ratings of each movie	10.369
Average rating of the movies	3.502
Maximum possible rating	5.0
Minimum possible rating	0.5
Median of the ratings	3.5

The dataset contains total ten possible ratings ranging from 0.5 to 5. Where rating 5 is considered as highest rating and 0.5 is considered as least rating. The difference between two closest rating is 0.5. Rating 4 is given by the users' maximum number of times. Table 2 gives the rating distribution in the dataset.

Table. 2: Rating distribution in MovieLens dataset.

Rating	Number of ratings
0.5	1370
1	2811
1.5	1791
2	7551
2.5	5550
3	20047
3.5	13136
4	26818
4.5	8551
5	13211

3.2 Attack Detection Attributes

The user profile possesses distinct properties that enable the differentiation between authentic user profiles and those associated with malicious intent. User profiles can be segregated based on various attributes, including but not limited to:

i. **Degree similarity with top k neighbors ($DegSim_u$):** The calculation of the similarity between User u and User v is determined according to Equation 1. The calculation of the average similarity between the k nearest neighbors and user u is performed according to equation number 3..

$$DegSim_u = \frac{\sum_{v=1}^{k} sim_{u,v}}{k} \tag{3}$$

Here, $sim_{u,v}$ is the similarity between user *u* and user *v*.

ii. **Length variance ($LengthVar_u$):** The underlying principle of employing this attribute is generally based on the notion that a genuine user refrains from assigning ratings to a significant quantity of items. If a user intentionally submits an excessive number of ratings within the system in order to maximize its impact, the following scenario will occur. The assignment of a maximum or minimum rating to the target item, as well as the assignment of ratings to numerous other items, is determined by the system. These assignments are made based on the attack's property. One plausible hypothesis is that the user profile under consideration has failed to furnish genuine ratings for the items within the system. Hence, the system shall detect the profile as fraudulent and subsequently disregard any ratings linked to it within the system. The length variance attribute quantifies the degree of variation in the length of user u in relation to the average length of other users within the system. The equation "2" denotes the assigned numerical value.

$$LengthVar_u = \frac{|l_u - \bar{l}|}{\sum_{k \in U}(l_k - \bar{l})^2} \tag{4}$$

Here, l_u is the length of user profile *u* i.e. number of ratings given by the user *u* in the system. $\bar{l}$ is the average length of user profiles in the system.

iii. **Rating deviation from mean agreement (RDMA):** this attribute measures the average deviation in the ratings for all the items that has been rated by user *u*. It is given by the equation 5.

$$RDMA_u = \frac{\sum_{i=0}^{N_u} \frac{|r_{u,i} - \overline{r_i}|}{t_i}}{N_u} \tag{5}$$

Here, N_u is the number of ratings given by the user *u*. t_i is the number of ratings given by all the users to item *i*. $r_{u,i}$ is the rating given by the user *u* to item *i*.

iv. **Weighted deviation from mean agreement (WDMA):** it is calculated by the equation 6.

$$WDMA_u = \frac{\sum_{i=0}^{N_u} \frac{|r_{u,i} - \overline{r_i}|}{|t_i|^2}}{N_u} \tag{6}$$

The Procedures and Outcomes of the Experiments

In this research, attacks are implemented in the eight different scenarios as shown in the Table 3.

Table 3: Informed attack model's attack scenarios.

Attack Name	**Intention**	**Fixed Attribute** (Target Item Size)	**Variable Attribute** (Attack size)
Probe Attack	Push	1	1%, 4%, 8%, 12%, 16%, 20%)
		10	
	Nuke	1	
		10	
Power User Attack	Push	1	
		10	
	Nuke	1	
		10	

This paper presents a comparative analysis of the accuracy achieved by three unsupervised models: k-means, hierarchical clustering, and EM (expectation maximization). In order to evaluate the resilience of the model, a 10-fold cross validation technique is employed. The charts depict the mean accuracy for both push and nuke attack intentions. This methodology enables the representation of eight distinct attack scenarios using only four charts.

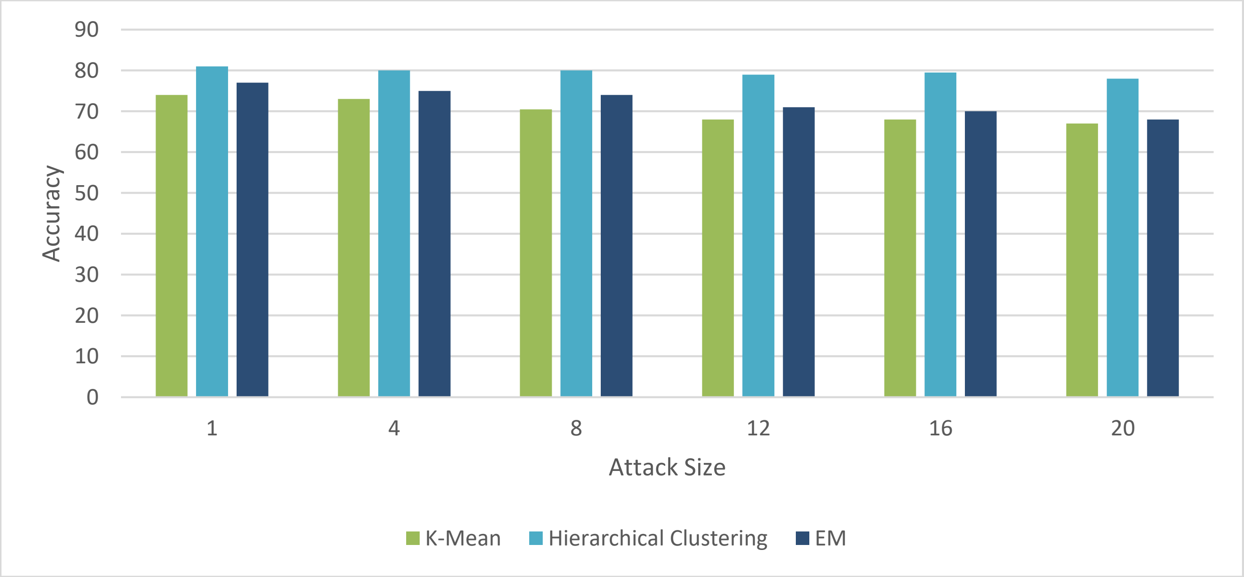

Fig. 1: Attack size versus average accuracy of push and nuke indentation of probe attack when attack size varies and target item size is 1.

Based on the findings presented in Figure 1, it has been determined that the hierarchical clustering model exhibits superior performance when compared to the other two models. The average accuracy of the hierarchical clustering model remains consistent at approximately 80% across various attack sizes. However, it is important to note that the average accuracy of all three models decreases as the attack size increases. The K-means model exhibits the lowest performance when compared to the other three models. The average accuracy gap between hierarchical clustering and the other two models also exhibits an increase as the attack size increases.

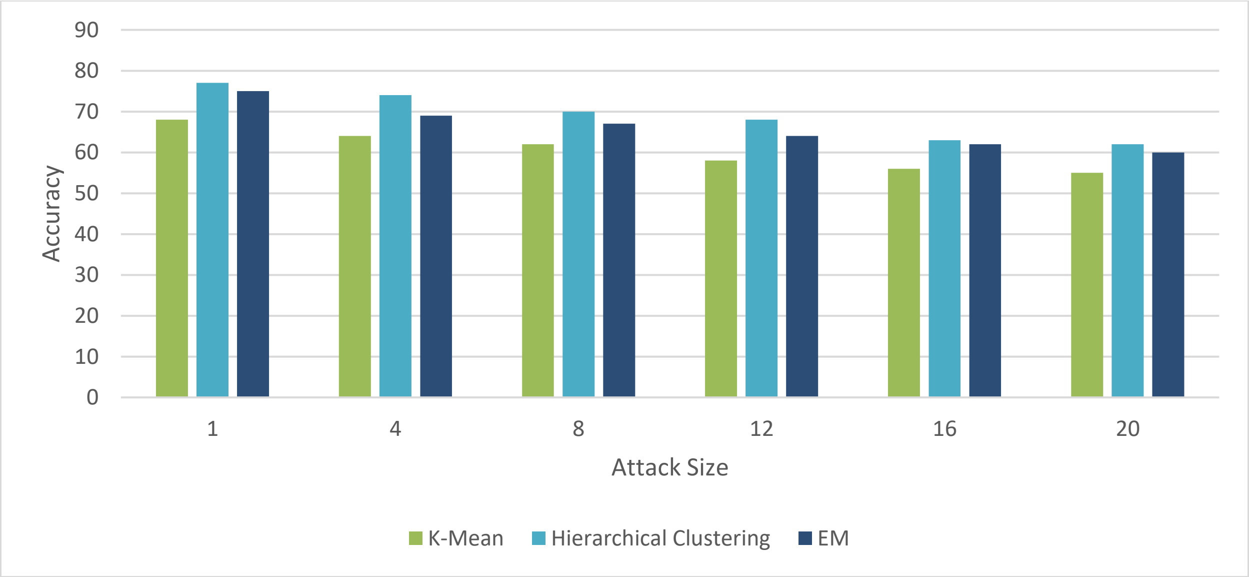

Fig. 2: Attack size versus average accuracy of push and nuke indentation of probe attack when attack size varies and target item size is 10.

In Figure 2, it can be observed that the average accuracy of all three models decreases in comparison to the scenario depicted in Figure 1, where the target item was 1. Additionally, it was observed that the performance gap among all three models is not significantly large. The average accuracy of hierarchical clustering and EM models exhibits a high degree of comparability when considering attack sizes of 8% or greater.

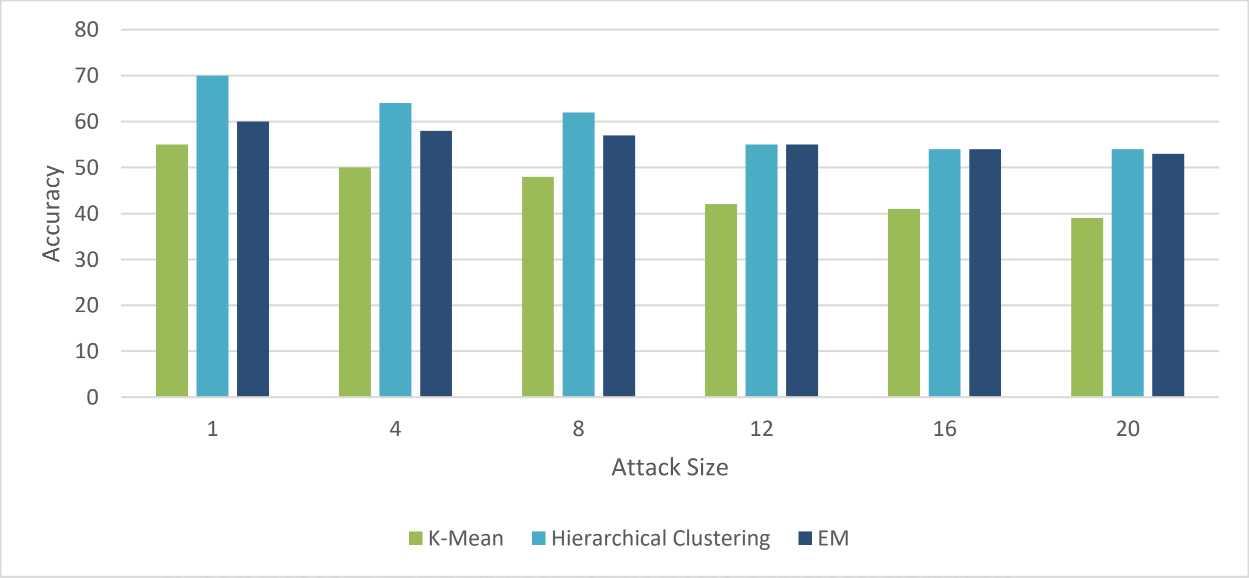

Fig. 3: Attack size versus average accuracy of push and nuke indentation of power user attack when attack size varies and target item size is 1.

In Figure 3, it is evident that the performance of all three unsupervised models exhibits a decrease when subjected to a power user attack, in contrast to the probe attack. The potential cause for this behavior could be attributed to the fact that in power user attacks, the attack user profiles exhibit a higher degree of correlation with the genuine user profiles in comparison to the user profiles observed in probe attacks. The hierarchical clustering models exhibit an average accuracy that surpasses the performance of the EM model by a margin of 10%. However, as the attack size increases, the discrepancy in performance between these models diminishes. It is worth noting that the performance gap between the k-means and hierarchical clustering models remains relatively consistent across all attack sizes. Following the attack, it has been observed that the average accuracy of both the EM algorithm and hierarchical clustering is nearly identical, with a margin of 12%.

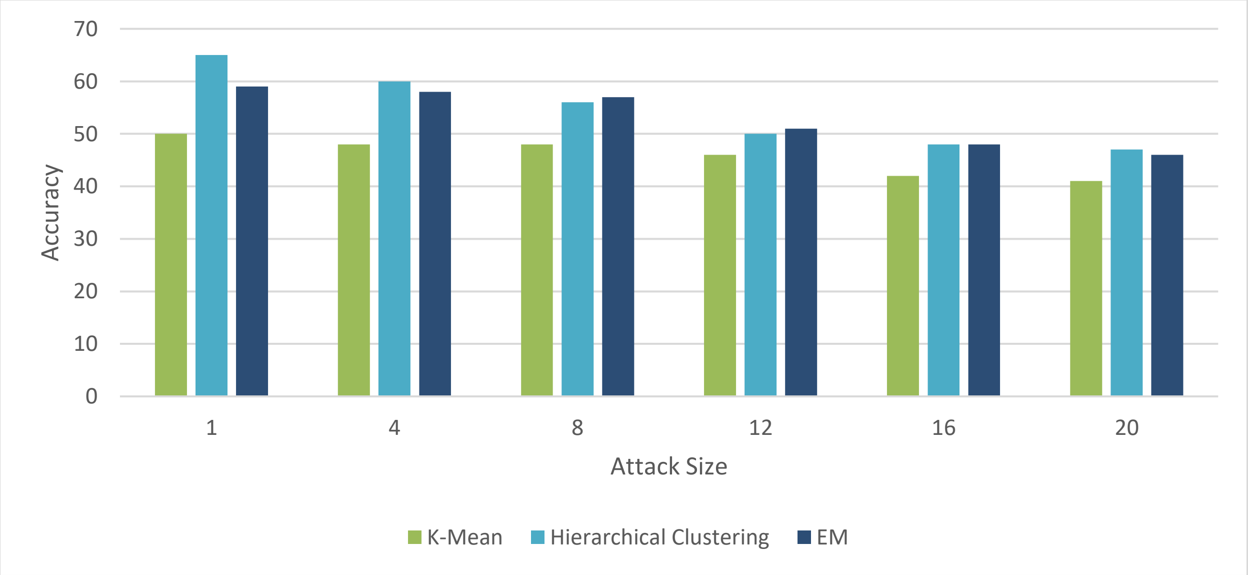

Fig. 4: Attack size versus average accuracy of push and nuke indentation of power user attack when attack size varies and target item size is 10.

In Figure 4, it has been observed that the hierarchical clustering model, which previously outperformed other models by a significant margin, is no longer the frontrunner. While the EM models initially outperform the hierarchical clustering model at an attack size of 1%, their performance becomes comparable as the attack size increases. Additionally, the k-means model narrows the performance gap with the other two models.

4. Conclusion and Future Scope

This paper examines two informed attacks and evaluates the average accuracy of three unsupervised machine learning models (k-means, hierarchical clustering, and EM) by varying the attack size from 1% to 20%. It has been observed that hierarchical clustering consistently outperforms the other two models across a wide range of scenarios. The K-means model exhibits the lowest performance compared to the other three models. The accuracy of these models exhibits a decline as the attack size increases.

The performance of these models can be enhanced through the exploration of novel attributes, thereby enabling further advancement of this research. This approach has the potential to be applied to other user profile attacks. The identification of novel profile injection attacks is possible, and subsequent application of unsupervised models can effectively address these attacks. Supervised machine learning models have the capability to be employed in the application of informed attack models. Ensemble models can be created by combining multiple supervised machine learning models, thereby enhancing model accuracy even in the most challenging scenarios.

5. References

[1]. Nguyen, Thi Thanh Sang, Hai Yan Lu, and Jie Lu. "Web-page recommendation based on web usage and domain knowledge." *IEEE Transactions on Knowledge and Data Engineering* 26.10 (2013): 2574-2587.

[2]. Panagiotakis, Costas, Harris Papadakis, and Paraskevi Fragopoulou. "Detection of hurriedly created abnormal profiles in recommender systems." 2018 International Conference on Intelligent Systems (IS). IEEE, 2018.

[3]. Bobadilla, Jesús, et al. "Recommender systems survey." Knowledge-based systems 46 (2013): 109-132.

[4]. Hameed, Mohd Abdul, Omar Al Jadaan, and Sirandas Ramachandram. "Collaborative filtering based recommendation system: A survey." International Journal on Computer Science and Engineering 4.5 (2012): 859.

[5]. Suganeshwari, G., and S. P. Syed Ibrahim. "A survey on collaborative filtering based recommendation system." Proceedings of the 3rd international symposium on big data and cloud computing challenges (ISBCC–16'). Springer, Cham, 2016.

[6]. Linden, Greg, Brent Smith, and Jeremy York. "Amazon. com recommendations: Item-to-item collaborative filtering." IEEE Internet computing 7.1 (2003): 76-80.

[7]. Burke, Robin, Bamshad Mobasher, Roman Zabicki, and Runa Bhaumik. "Identifying attack models for secure recommendation." In Beyond Personalization: A Workshop on the Next Generation of Recommender Systems. 2005.

[8]. O'Mahony, Michael, Neil Hurley, Nicholas Kushmerick, and Guénolé Silvestre. "Collaborative recommendation: A robustness analysis." ACM Transactions on Internet Technology (TOIT) 4, no. 4 (2004): 344-377.

[9]. Lam, Shyong K., and John Riedl. "Shilling recommender systems for fun and profit." In Proceedings of the 13th international conference on World Wide Web, pp. 393-402. ACM, 2004.

[10]. Zhang, Fuzhi, et al. "UD-HMM: An unsupervised method for shilling attack detection based on hidden Markov model and hierarchical clustering." Knowledge-Based Systems 148 (2018): 146-166.

[11]. Davoodi, Fatemeh Ghiyafeh, and Omid Fatemi. "Tag based recommender system for social bookmarking sites." 2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. IEEE, 2012.

[12]. Pitsilis, Georgios K., Heri Ramampiaro, and Helge Langseth. "Securing Tag-based recommender systems against profile injection attacks: A comparative study." arXiv preprint arXiv:1901.08422 (2019).

[13]. Ricci, Francesco, Lior Rokach, and Bracha Shapira. "Recommender systems: introduction and challenges." Recommender systems handbook. Springer, Boston, MA, 2015. 1-34.

[14]. Lops, Pasquale, Marco De Gemmis, and Giovanni Semeraro. "Content-based recommender systems: State of the art and trends." Recommender systems handbook (2011): 73-105.

[15]. Su, Xiaoyuan, and Taghi M. Khoshgoftaar. "A survey of collaborative filtering techniques." Advances in artificial intelligence 2009 (2009).

[16]. Hill, Will, et al. "Recommending and evaluating choices in a virtual community of use." Proceedings of the SIGCHI conference on Human factors in computing systems. 1995.

[17]. Konstan, Joseph A., et al. "Grouplens: Applying collaborative filtering to usenet news." Communications of the ACM 40.3 (1997): 77-87.

[18]. Isinkaye, Folasade Olubusola, Yetunde O. Folajimi, and Bolande Adefowoke Ojokoh. "Recommendation systems: Principles, methods and evaluation." Egyptian informatics journal 16.3 (2015): 261-273.

[19]. Anwar, Taushif, and V. Uma. "A study and analysis of issues and attacks related to recommender system." Convergence of ICT and Smart Devices for Emerging Applications. Springer, Cham, 2020. 137-157.

[20]. Cohen, R., Sar Shalom, O., Jannach, D., & Amir, A. (2021, March). A Black-Box Attack Model for Visually-Aware Recommender Systems. In Proceedings of the 14th ACM International Conference on Web Search and Data Mining (pp. 94-102).

[21]. Domingos, Pedro, and Matt Richardson. "Mining the network value of customers." Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining. 2001.

[22]. Fang, Minghong, Neil Zhenqiang Gong, and Jia Liu. "Influence function based data poisoning attacks to top-n recommender systems." Proceedings of The Web Conference 2020. 2020.

[23]. Rashid, Al Mamunur, George Karypis, and John Riedl. "Influence in ratings-based recommender systems: An algorithm-independent approach." Proceedings of the 2005 SIAM International Conference on Data Mining. Society for Industrial and Applied Mathematics, 2005.

[24]. Panagiotakis, Costas, Harris Papadakis, and Paraskevi Fragopoulou. "Unsupervised and supervised methods for the detection of hurriedly created profiles in recommender systems." International Journal of Machine Learning and Cybernetics 11.9 (2020): 2165-2179.

[25]. MovieLens homepage: https://grouplens.org/datasets/movielens/.

[26]. Chirita P, Nejdl W, Zamfir C (2005) Preventing shilling attacks in online recommender systems. In: WIDM ’05: Proceedings of the 7th annual ACM international workshop on Web information and data management, New York, NY, USA, ACM Press pp 67–74