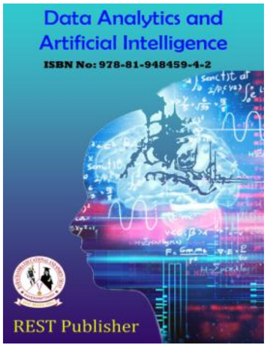
Data Analytics and Artificial Intelligence

Vol: 3(2), 2023

REST Publisher; ISBN: 978-81-948459-4-2

Website: http://restpublisher.com/book-series/daai/

DOI: https://doi.org/10.46632/daai/3/2/17


Data Analytics and Artificial Intelligence
ISBN No: 978-81-948459-4-2
REST Publisher

Shilling Attack Detection in User Based Recommendation System

*** S. Poornima, M. Geethanjali**

St. Joseph's College of Arts and Science for Women, Hosur, Tamil Nadu, India

*Corresponding Author Email: poornimadazzling01@gmail.com

Abstract. *The majority of the existing unsupervised methods for detecting shilling attacks are based on user rating patterns, ignoring the differences in rating behavior between legitimate users and attack users. These methods have low accuracy in detecting different shilling attacks without having any prior knowledge of the attack types. We provide a novel unsupervised shilling assault detection technique based on an examination of user rating behavior in order to overcome these constraints. By first examining the deviation of rating tendencies on each item, we are able to determine the target item(s) and the accompanying goals of the attack users. Based on the results of this study, a group of suspicious users is then created. Second, we examine the users' rating behaviors in terms of their rating and interest preferences. Finally, using measurements of user rating behavior, we determine the suspicious degree and identify attack users within the collection of suspicious users. The Movie Lens 1M dataset, the sampled Amazon review dataset, and the Netflix dataset all show how good the suggested detection model.*

Keywords: *Recommender systems, Shilling attacks, Shilling attack detection, Target item identification, User rating behavior analysis (AURB), density-based clustering algorithm.*

1. INTRODUCTION

Recommender systems have been widely used in many industries to address the issue of information overload. Examples include product recommendations on e-commerce websites, webpage recommendations in intelligent Web systems, and POI (point-of-interest) recommendations in location-based social networks. One of the most popular methods in recommender systems is collaborative filtering (CF), which has become a crucial component of many e-commerce websites like Amazon and Netflix. However, fraudulent individuals may introduce numerous false profiles into a CF recommender system in an effort to alter the frequency at which a specific item is recommended, often motivated by financial gain. Shilling attacks or profile injection attacks have been used to describe this behavior. Shilling assaults can be classified into several categories depending on their purpose. Numerous techniques to identify such attacks have been proposed in an effort to lessen the impact of shilling attacks on CF recommender systems. From the perspective of machine learning, the current shilling attack detection techniques can be divided into supervised, semi-supervised, and unsupervised methods. Unsupervised detection techniques have received a lot of interest because they do not require classifier training and are effective regardless of the type of assault. These techniques involve prior attack knowledge and mostly concentrate on the rating patterns of the attack users (e.g., the attack size).However, attack users may mimic real users' rating behaviors (for example, by rating a number of popular goods like real users do), thus in fact, previous awareness of assaults is not required.

2. RELATED WORKS:

Techniques have been proposed, which can be divided into supervised, semi-supervised, and unsupervised techniques. Williams et al presented certain generic and model-specific features and trained three supervised classifiers to detect common attacks (i.e., random attack, average attack, and bandwagon attack) under the category of supervised detection approaches. In order to choose the most useful detection features for well-known forms of assaults, Wu et al introduced a feature selection technique. In order to detect different assaults, Yang et al introduced a rescaled AdaBoost based on 18 statistical features of the attackers. This method surpassed baseline methods in recognizing known forms of attacks. However, these qualities demand a significant amount of computing work. Using an SVM-based classifier and target item analysis, Zhou et al offered a two-stage method that first employed a Borderline-SMOTE method to address the class unbalance problem, and then applied a target item analysis. Wu et al. suggested a semi-supervised method for distinguishing genuine and attack profiles using both labelled and unlabeled data in the area of semi-supervised detection methods. A classifier was first trained on a small batch of labelled data, and the initial classifier was subsequently enhanced by include a weighting factor for EM. A semi-supervised technique was recently used by Zhang et al.to identify spammer groups from product reviews. In order to detect recognized types of attacks, semi-supervised detection techniques still need to train classifiers using certain labelled profiles. s. They first constructed an undirected user-user graph, and then calculated the similarity of the topological structure of the attack users in a graph model and finally captured the attack users by target item analysis. This method can detect shilling attacks with various attack sizes and filler sizes but determining the empirical parameters is crucial to its detection performance. Recently, Zhang et al. used a hidden Markov model and hierarchical clustering to detect shilling attacks in recommender systems. This method performed very well in detecting most types of shilling attacks, but it suffered from low precision in detecting the AoP attack. In this research, we focused on creating a group of suspect users and examining the behavioral distinctions between attack users and legitimate users. Our methodology does not require prior information of the attack size in advance, in contrast to the unsupervised detection methods in. In contrast to unsupervised detection method, our method does not need labelling the seed spam users. Our methodology, which increases the accuracy of identifying the target items as compared to the method in identifies the target item(s) by leveraging the variance in the rating distributions of the target items. Comparing the distinctions between real users and attack users in their interest and rating preferences with that of our approach achieves.

3. METHODOLOGIES:

Our Proposed System Includes

- Using the deviation of rating distributions on items, we identify the target item(s) and attack intent, and based on this, we create a list of suspect users, which includes all attack users as well as some real users.
- In order to construct ordered user interest preference sequences and characterize the behavior difference in interest preference between genuine and attack users, we obtain the latent classes of each item. By doing so, we can measure the variety and memory of user interest preference using information entropy.
- We create ordered user rating sequences and evaluate memory of rating preference using a self-correlation technique to define the behavior difference in rating preferences between genuine users and attack users.
- We determine each user's suspicious degree based on the extracted behavior features (i.e., variety of interest preference, recall of previous actions, etc.) to separate attack users from legitimate users in the group of suspicious users. A density-based clustering technique is used to identify users (by spot attack users, interest preference, and memory of rating preference).

Framework of DSA-AURB: The DSA-detection AURB's framework as shown in Fig.1illustrates the three stages of the DSA-AURB process: creating the list of suspect users, extracting behavior data, and identifying assaulting users. A group of suspect users is created based on the identified target items after we first establish an ordered rating sequence for each item and identify the target item(s). Aspects of each user's interest preferences and rating preferences are used to extract three behavior traits in the second stage after the ordered item sequence and rating sequence for each user in the collection of suspicious users are formed. The suspicious degree of each user in the group of suspicious users is determined in the third stage.

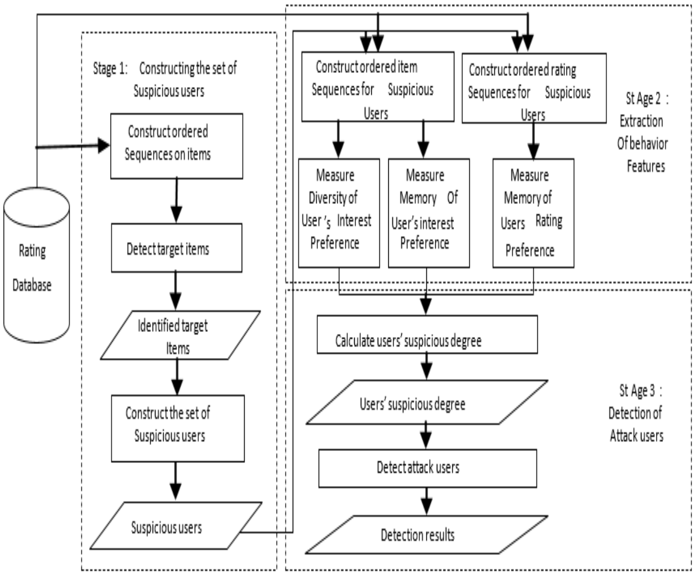

Stage 1: Constructing the set of Suspicious users
Construct ordered Sequences on items
Rating Database
Detect target items
Identified target Items
Construct the set of Suspicious users
Suspicious users
Construct ordered item Sequences for Suspicious Users
Construct ordered rating Sequences for Suspicious Users
St Age 2 : Extraction Of behavior Features
Measure Diversity of User's Interest Preference
Measure Memory Of User's interest Preference
Measure Memory of Users Rating Preference
Calculate users' suspicious degree
Users' suspicious degree
Detect attack users
Detection results
St Age 3 : Detection of Attack users

FIGURE 1. Framework of DSA-AUR\\B

Target Items Analysis: Attackers frequently give target items high ratings for push attacks or low ratings for nuke attacks, which results in a multitude of aberrant ratings for the target items within a comparatively short time. We make the same assumption as the research in that, despite the variety of user ratings, the rating distributions of normal objects are often stable and do not fluctuate dramatically over time. Therefore, by examining items with a significant shift in the rating distribution, the target items can be found. Each item is first given an ordered rating sequence based on its rating time, which is then separated into a number of distinct rating windows. We next estimate the location of the abnormal rating window and adaptively compare the similarity of rating distributions using the Hellinger distance. Each item is first given an ordered rating sequence based on its rating time, which is then separated into a number of distinct rating windows. s. We next estimate the location of the abnormal rating window and adaptively compare the similarity of rating distributions using the Hellinger distance. Database of ratings Create organized lists of objects Discover target objects target goods identified Create a list of suspicious users. Building the list of questionable users is the first stage. Make organized item sequences for users who seem suspicious. Create sequential ratings for questionable users. Determine the variety of consumer interest preferences. Measure user interest preference memory Measure user rating preference memory

Stage1: Extracting behavioral characteristics. Determine the level of suspicion for users' level of suspicion. Identify the attackers. Results of detection

Stage2: Attack users alter the abnormal window's size after being detected. The outlier identification approach is then employed to identify.

Algorithm 1: Adjusting the window size of attacks on the ordered rating sequence of item pi

Input: ORSIi, window size 0 k, the starting and final positions for $SusWin_i$

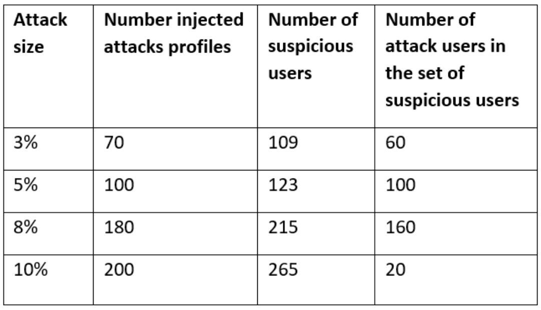
Output: the starting and final positions for attack rating

```
Beginp ← the starting position for $SusWin_i$
Endp ← the final position for $SusWin_i$
$RP_{win}$ rating distribution on the window $SusWin_i$
flag1 ← 1
Size ← $k_o$
while flag1 do
if beginp-size 1 then
```

```
 RP f ← rating distribution from beginp-size to endp
**if** H( ) RP win RP f )< then
beginp ← ( beginp-size )
**else** size ← size / 2
**end if**
**if** i endp n or 0 size k /16 then
 flag2 0
**end if**
 **end while**
 **end if**
```

Construction of suspicious users set: Let Susp Item Set represent the collection of all found target items. Regarding pi Based on the shift in the rating distributions on the ORSIi, SuspItemSet, we can infer intenti. In the case of a push assault, for instance, attack users rate the target item with a maximum of rmax (for example, random attack and average attack) or rmax -1 (for instance, average-target shift attack). High ratings (rmax and rmax 1) on the target item are regarded as suspicious ratings, hence intenti is set to 1 if the ratings on the anomalous window of ORSIi focus on high ratings. In the case of a nuke attack, attack users assign rating rmin (for example, a love/hate attack) or rmin +1 (for example, an average-target shift attack) to the target object, therefore intenti is set to -1.

TABLE 1. Number of attack users included in the set of suspicious users for various attack sizes



Attack size	Number injected attacks profiles	Number of suspicious users	Number of attack users in the set of suspicious users
3%	70	109	60
5%	100	123	100
8%	180	215	160
10%	200	265	20

Algorithm 2: Constructing the set of suspicious users

Input: rating time matrix T, rating matrix R and k0

Output: the set of suspicious users SuspUserSet (1) SuspUserSet ←ϕ and SuspItemSet ←ϕ

```
**for**∀∈p I i **do**
ORSIi ← sort all the ratings on item pi according to their timestamps in ascending order
**if** i 0 n k ≥
0 i n ws k⌈ ⌉= ⌊⌋ // to compute the number of time windows on item pi
**for** win=1 to ws do
RPwin ← get the rating distribution of window win
compute H( , ) RP RP win
**end for**
**end if**
**for** i ∀∈p SuspItemSet do
```

```
for k=1 to U do
    if ((intenti=1) and ( r r ki ≥– max 1 )) or ((intenti=1) and ( r r ki ≤+ min 1 )) then
        add user uk to SuspUserSet
    end if
end for
end for
return SuspUserSet
```

Behavior Features Extraction We employ a probabilistic latent semantic analysis (PLSA) to determine the topic distributions of each item in order to examine a user's preferred areas of interest. Let k be the quantity of concealed topics (also referred to as the latent class in this study), and 1 2 {...} be the collection of all latent classes, Z z z z k. There may be a covert association between an item and numerous latent classes. The weight of the relationship between item pi and the latent class zj is shown by the value of (|) j I p z p, which is represented as (1,2,...,, 1,2,...,) w I n j k I j and, 1 1 k I j j w . The greater wi j is, the more closely the two are related.

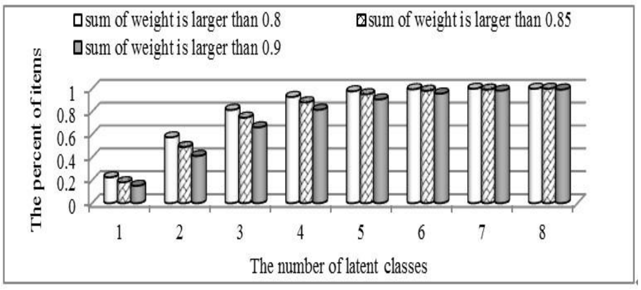


FIGURE 2. Relationship between length of vector and sum of weights

Memory of User’s Interest Preference: Users have a tendency to spend a lot of time on particular interest preferences, according to research in. In addition words, the majority of typical users have recurring interests and habits. This indicates that a user's interest preferences may be stored in an ordered sequence in memory. As a result of the study in, we make use of the notion of block entropy to assess a user's recollection of their preferred areas of interest.

Memory of user’s rating preference: According to behavioral psychology research, anchoring effects have an impact on behavioral decisions, which indicates that people frequently look back at past conduct when selecting how to act in the present. Previous research has shown that even a randomly chosen initial value for an object can have a significant impact on how its genuine worth is estimated. People are more inclined to rate something poorly in recommender systems if they previously gave it a poor rating, and vice versa. In other words, the rating we give an object now could be influenced by the rating we gave it earlier because we might utilize that earlier rating as an anchor. As a result, we apply the anchoring effect to measure user rating preference memory in this section.

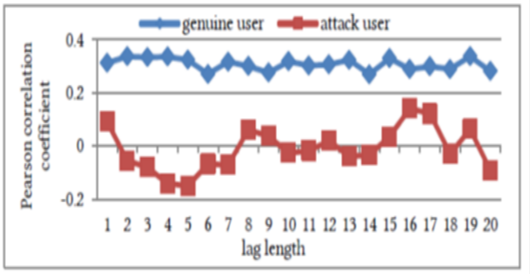


FIGURE 3. Relation between the Person correlation coefficient and the lag length

Algorithm 3: Detecting attack users based on clustering

Input: ItemClass, rating time matrix T and rating matrix R

Output: the set of attack users ShillingSet

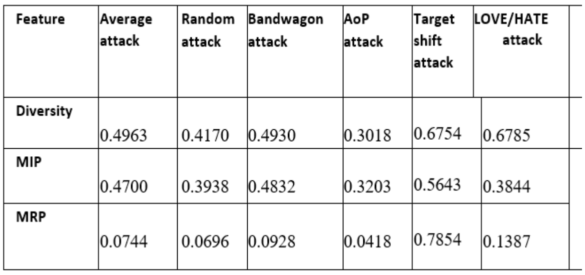
```
ShillingSet=ϕ
SusUserSet ← call Algorithm 2
for each user u in SusUserSet do
compute Diversity(u)
compute MIPu
compute MRPu
end for
maxD ← the maximum of users’ interest preference diversity in SusUserSet
maxMIP ← the maximum of users’ real entropy in SusUserSet
minMIP ← the minimum of users’ real entropy in SusUserSet
for each user u in SusUserSet do
calculate Suspu
end for
```

4. EXPERIMENT AND ANALYSIS

We use the Movie Lens 1M dataset as the experimental data in order to assess the performance of DSA-AURB. Movie Lens 1M dataset, second. This dataset includes 1,000,209 ratings for 3952 films from 6040 individuals. Every rating is an integer number between 1 (disliked) and 5. (Most liked). At least 20 films have been rated by each user. All profiles on the Netflix and Movie Lens 1M datasets are taken to be real profiles, similar to other work on synthetic datasets. The attack profiles are produced utilizing average attack, random attack, bandwagon attack, AOP attack, average-target shift attack, average-noise injecting assault, PUA, love/hate attack, and reverse bandwagon attack, respectively, to simulate a variety of attacks in the recommender systems. Push attacks are carried out using the first seven attack models, while nuclear attacks are carried out using the final two attack models. Attackers want to alter the recommendations of their target objects by pushing them up or down (push assaults) or nuking them altogether (nuke attacks). The experiment's target object is selected using the following techniques: For push attacks, one of the items is randomly chosen as the target. Due to the attack impacts and costs, attack profiles are typically injected quickly in practice. As a result, in our trials, attack users' rating timestamps are widely dispersed. The filler size is not taken into account for the PUA because the attackers' filler items are chosen based on those that real users have given high ratings. We used in degree and Number Ratings, two separate heuristic techniques for power user identification. PUA-In degree and PUA-Number Ratings signify the corresponding attack using the two techniques of power user identification, respectively, to distinguish the two approaches for power user identification. Each of our investigations is run ten times, with the average of the detection results serving as the final assessment.

Analysis of information gain: We determine each dataset's information gain for each feature utilized in Stage 2 before presenting it. Which of the traits are most crucial for a categorization system can be determined via information gathering. The importance of the feature increases with the size of the information gain. The suggested features on three datasets, where attack profiles are produced by various attacks with 5% attack size and 5% filler size, respectively, on the Netflix and Movie Lens 1M datasets. A 10-trial average is used for all outcomes. The Movie Lens 1M datasets, (i.e., variety of user's interest preference) and MIP (i.e., true entropy of user) are two crucial variables for detecting various assaults, showing that attack profiles and genuine profiles have notable variations.

TABLE 2. Information gain of the proposed features on the Movie lens 1M dataset.



Feature	Average attack	Random attack	Bandwagon attack	AoP attack	Target shift attack	LOVE/HATE attack
Diversity	0.4963	0.4170	0.4930	0.3018	0.6754	0.6785
MIP	0.4700	0.3938	0.4832	0.3203	0.5643	0.3844
MRP	0.0744	0.0696	0.0928	0.0418	0.7854	0.1387

5. EXPERIMENTAL RESULT AND ANALYSIS:

In order to evaluate the viability of the suggested strategy, we contrast DSA-AURB with the following four benchmark techniques.

- **PCA-Reselect:** Mehta et AL traditional unsupervised shilling attack detection algorithm, which performs well if given information about the amount of attack users. In the experiments, we presumptively know the attack size beforehand.
- **CBS (short for Catch the Black Sheep):** unified framework for shilling attack detection based on fraudulent action propagation) A ranking method based on fraudulent action propagation that determines the spam probability for each user and each item and considers the top-k users as the attack users.
- **EUB-DAR (short for "Estimating User Behavior toward Detected Anomalies in Rating Systems"):** Yang et AL unsupervised. Shilling attack detection technique, it uses target item analysis and the similarity of the topological structures of the attack users in a graph to identify the attackers. The parameters t and are set to 30 and 20, respectively, in the experiments.
- **UD-HMM:** This unsupervised shilling attack detection technique relies on hierarchical clustering and a hidden Markov model, and it often performs very well in identifying a variety of assaults. N and set to 5 and 0.7, respectively, in the experiments.

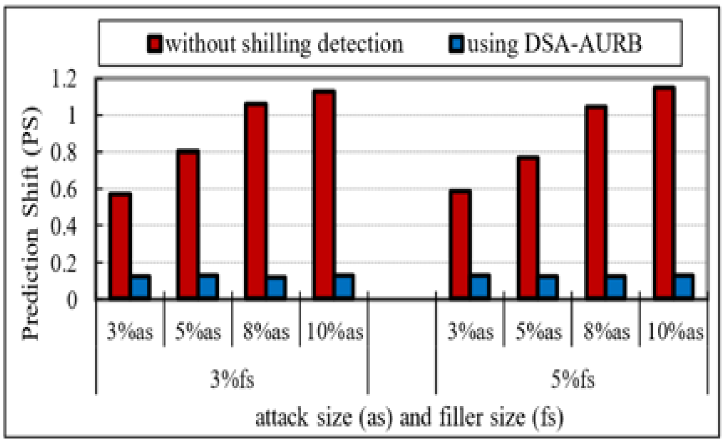
Comparison of detection results for five methods on the Movie Lens 1M dataset: With the exception of bandwagon attack and AoP attack, PCA-Var Select detects nine attacks with a detection precision of between 0.8914 and 0.9457 in the Movie Lens 1M dataset. This result suggests that PCA-Var Select can successfully identify the seven attacks when it is aware of the attack size. However, PCA-detection Var Select precision is poor under bandwagon attack and AoP attack, indicating that it is unable to identify the attack profiles produced by these attacks. The detection accuracy of CBS under different threats ranges from 0.8876 to 0.9704. This indicates that if the number of attack users is known in advance and a number of attack users are chosen as seed, CBS performs very well in detecting attack profiles generated by different attacks. This observation was made mostly due to the similar item popularities between attack profiles created by AoP attacks and real profiles. The DSA-detection AURB's precision values under a variety of attacks with a variety of attack sizes and filler sizes, are nearly all 100%. This indicates that DSA-AURB can accurately identify attack characteristics for a variety of attacks. These findings also suggest that the types of attacks have no impact on the detection accuracy of DSA-AURB. Consequently, DSA-AURB has the best precision under multiple attacks on the Movie Lens 1M dataset when compared to the four benchmark methods. Among the five approaches tested using the Movie Lens 1M dataset, DSA-AURB had the highest F1-measure values for detection under a variety of attacks with varying attack and filler sizes. Under bandwagon attack and AoP attack, the F1-measure of PCA-Var Select is low. Additionally, when compared to the other seven attacks with different attack sizes and filler sizes, the F1-measure of PCA-Var Select is less favorable than that of DSA-AURB. The F1-measure values of CBS are high under different attacks, demonstrating that the type of attack has no bearing on CBS's ability to detect threats. However, compared to DSA-AURB, its F1 measure is lower. Under push attacks with small attack sizes (e.g., 3% attack size) and small filler sizes (e.g., 3% filler size), the F1-measure of EUB-DAR is not high. Outstanding F1-measure of UD-HMM.

Comparison of detection results for the five methods on the sampled Movie lens review dataset: On the sampled Amazon review dataset, the detection precision of the five algorithms is not very great. The fact that most attack users only give ratings to target products and most genuine users rate a huge range of unpopular items could be the cause. As a result, a sizable proportion of real users are mistakenly classified as attack users by these identification techniques. The table shows that DSA-detection AURB's recall is 0.9630, which is higher than that of the other four benchmark approaches. The fact that the suspicious degrees for attack users are greater and closer together than those for many genuine users suggests that our approach can detect the majority of attack users.

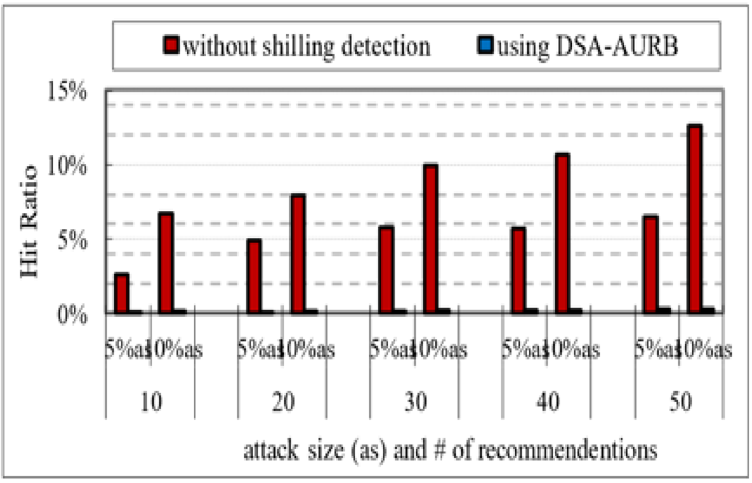
Method	precision	recall	F1-measure
PCA-VarSelect	0.3245	0.6785	0.4567
CBS	0.6754	0.3245	0.4786
EUB-DAR	0.3456	0.5678	0.2346

UD-HMM	0.4563	0.5678	0.6754
DSA-AURB	0.3883	0.9630	0.5534

Case Scenario in the Recommendation Algorithm: We measure the stability of the recommendation algorithm on the movie lens dataset using prediction shift (PS) and hit ratio metrics to show whether the proposed detection approach can help to improve the robustness of recommender systems. The average prediction shift and hit ratio for the pushed item increase with the attack size in the case without shilling detection, and the average prediction shift decreases to a small value by using DSA-AURB. Using DSA-AURB, the hit ratio findings are close to zero regardless of attack magnitude. These findings suggest that DSA-AURB can help make recommender systems more resilient to shilling attacks.



(a) Prediction shift



(b) Hit ratio

6. CONCLUSION AND FUTURE ENHANCEMENT

We propose a novel unsupervised shilling attack detection model based on analysis of user rating behavior, which makes use of the target item(s) and the differences in rating behavior between genuine and attack users in recommender systems, in order to enhance the performance of unsupervised shilling attacks detection methods with no prior knowledge of attacks. We provide a method of distance-based analysis to locate the target item based on the presumption of steady rating patterns on things without attacks (s). We create a group of suspect users, which includes attack users and some real users, based on the target items that have been discovered. Instead of looking at the entire system's user base, we focus on the set of suspect users, which simplifies the computation. We measure the user suspicious degree for each user in the list of suspicious users in order to find attack users among them. We use PLSA to determine each item's latent classes; from there, the ordered interest preference sequences are generated. We employ information entropy and block entropy to assess the diversity and memory of user interest preferences based on the generated interest preference sequences. We create the ordered rating sequences, and as a result, the anchoring effect may be used to describe how user rating preferences are remembered. These behavioral traits can be used to determine a user's level of suspicion and identify users who are under attack. The Movie Lens 1M dataset, the sampled Amazon review dataset, and experimental results on the Netflix dataset have all demonstrated the exceptional performance of DSA-AURB in detecting various shilling attacks. We'll present a more adaptable approach to recognize different target items in **further work.** Additionally, greater investigation will reveal the hidden connections between people and things in recommender systems and introduce new group features for identifying attack users.

ACKNOWLEDGEMENT:

I am delighted to express my heartfelt appreciation to our department’s head and staff, as well as family and friends. This paper is made possible by their encouragement, assistance, and support.

REFERENCES

[1]. H. Yin, X. Zhou, B. Cui, et al., Adapting to user interest drift for POI recommendation, IEEE Transactions on Knowledge and Data Engineering, 28 (10) (2016) 2566-2581.
[2]. B. Mobasher, R. Burke, R. Bhaumik, et al., Attacks and remedies in collaborative recommendation, IEEE Intelligent Systems, 22 (3) (2007) 56-63.
[3]. W. Li, W. Pedrycz, X. Xue, et al., Distance-based double-quantitative rough fuzzy sets with logic operations, International Journal of Approximate Reasoning, 101(2018) 206-233.
[4]. Z. Liang, W. Li, Y. Li, A parallel Probabilistic Latent Semantic Analysis method on MapReduce platform, IEEE International Conference on Information and Automation, IEEE, 2014, pp. 1017-1022.
[5]. S. Jahirabadkar, P. Kulkarni, Algorithm to determine ε-distance parameter in density based clustering, Expert Systems with Applications, 41(6) (2014) 2939–2946.
[6]. C. A. Williams, B. Mobasher, R. Burke, et al., Detecting profile injection attacks in collaborative filtering: a classification-based approach, Knowledge Discovery on the Web International Conference on Advances in Web Mining and Web Usage Analysis, Springer-Verlag, 2006, pp. 167-186.

[7]. Z. Wu, Y. Zhuang, Y. Wang, et al, Shilling attack detection based on feature selection for recommendation system, ACTA ELECTRONICA SINICA, 40(8) (2012) 1687−1693 (in Chinese).
[8]. F. Zhang, H. Chen, An ensemble method for detecting shilling attacks based on ordered item sequences, Security and Communication Networks, 9 (2016) 680-696.
[9]. Q. Zhou, Supervised approach for detecting average over popular items attack in collaborative recommender systems, IET Information Security, 10(3) (2016) 134-141.
[10]. J. Lee, D. Zhu, Shilling attack detection - A new approach for a trustworthy recommender system, Informs J Comput, 24 (2012) 117-131.
[11]. J. Zou, F. Fekri, A belief propagation approach for detecting shilling attacks in collaborative filtering, Conference on Information & Knowledge Management, ACM, 2013, pp. 1837-1840.
[12]. Bilge, Z. Ozdemir, H. Polat, A novel shilling attack detection method, Procedia Computer Science, 31 (2014) 165-174.
[13]. Z. Zhang, S.R. Kulkarni, Detection of shilling attacks in recommender systems via spectral clustering, Proceedings of 17th International Conference on Information Fusion, IEEE, 2014, pp. 1-8.
[14]. Z. Yang, Z. Cai, X. Guan, Estimating user behavior toward detecting anomalous ratings in rating systems, Knowledge-Based Systems, 111 (2016) 144-158.
[15]. F. Zhang, Z. Zhang, P. Zhang, 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.