International Journal of Engineering

Journal Homepage: www.ije.ir

A New Mechanism for Detecting Shilling Attacks in Recommender Systems Based on Social Network Analysis and Gaussian Rough Neural Network with Emotional Learning

R. Moradi, H. Hamidi*

Department of Industrial Engineering, K. N. Toosi University of Technology, Tehran, Iran

P A P E R I N F O

Paper history:
Received 18 September 2022
Received in revised form 04 December 2022
Accepted 06 December 2022

Keywords:
Recommender System
Shilling Attack
Collaborative Filtering
Fake Profiles
Social Network

A B S T R A C T

A recommender system is an integral part of any e-commerce site. Shilling attacks are among essential challenges in recommender systems, which use the creation of fake profiles in the system and biased rating of items, causing the accuracy to decrease and the correct performance of the recommender system in providing recommendations to users. The target of attackers is to change the rank of content or items corresponded to their interests. Shilling attacks are a threat to the credibility of recommender systems. Therefore, detecting shilling attacks it necessary to in recommender systems to maintain their fairness and validity. Appropriate algorithms and methods have been so far presented to detect shilling attacks. However, some of these methods either examine the rating matrix from a single point of view or use low-order interactions or high-order interactions. This study aimed to propose a mechanism using users' rating matrix, rating time, and social network analysis output of users' profiles by Gaussian-Rough neural network to simultaneously use low-order and high-order interactions to detect shilling attacks. Finally, several experiments were conducted with three models: mean attack, random attack, and bandwagon attack, and compared with PCA, Semi, BAY, and XGB methods using precision, recall, and F1-Measure. The results indicated that the proposed method is more effective than the comparison methods regarding attack detection and overall detection, which proves the effectiveness of the proposed method.

doi: *10.5829/ije.2023.36.02b.12*

1. INTRODUCTION

A recommender system can be defined as a program that recommends appropriate items by predicting user preference for an item based on information related to items, users, and interaction between the two [1]. In the past 25 years, the personalization of e-services by recommender systems has received much attention [2, 3]. The growing importance of the web as a medium for electronic and commercial transactions has created a strong impetus in the development of recommender systems. One of the key factors in this regard is that the web allows users to provide feedback about their taste [4].

One of the challenges of life today is making the right choice when buying a product. This challenge is compounded due to the increasing volume, variety and velocity of product-related data [5]. Although the vast increase in the number of options has given consumers the opportunity to choose the most interesting products; it has also caused choice overload. This problem occurs when there are an infinite number of options to choose from that do not significantly differ from each other [6].

Recommender systems are primarily developed and integrated into e-commerce websites and have largely been able to help users make decisions. However, recommender systems have found applications beyond e-commerce websites and are used in almost every field from social networks to medical science [7, 8].

Recommender systems have improved user decisions when interacting with the system, and their effectiveness has been proven. For example, recommender systems

*Corresponding Author Institutional Email: h_hamidi@kntu.ac.ir (H. Hamidi)

Please cite this article as: R. Moradi, H. Hamidi, A New Mechanism for Detecting Shilling Attacks in Recommender Systems Based on Social Network Analysis and Gaussian Rough Neural Network with Emotional Learning, ***International Journal of Engineering, Transactions B: Applications***, Vol. 36, No. 02, (2023), 321-334

allow users to discover surprising items that may be unknown to them by receiving recommendations from unexplored parts of goods [9]. For this purpose, recommender systems observe user behaviors carefully and collect different forms of users to understand users' personal preferences [10, 11].

Recommender systems are currently used in various areas where there are various options to choose from, such as watching movies, reading books, buying goods, listening to music, visiting tourist areas, eating at restaurants [12].

Research on recommender systems was initially focused on the goal of providing accurate recommendations, but now, other goals such as novelty, diversity, reliability, etc., have also emerged beyond accurate recommendations [13]. Recommender systems are a powerful personalization tool that uses user behaviors to provide personalized options or adapt the user interface [2]. Currently, two factors, high dependence of users on recommender systems and the great interest of companies to provide user-friendly recommendations, have contributed to the success of recommender systems [12].

Providing recommendations in recommender systems in a set of cases such as products, business services and news can lead to significant changes such as increasing business profits or influencing public opinion. Due to the great importance of these systems, there is a strong interest in influencing the output of recommender systems. Part of the efforts to influence the output of recommender systems are done through legal and authorized actions such as advertising, enriching the information of the presented items, but another part is carried out by using illegal and deceptive actions such as attacking the recommender systems.

Collaborative filtering based recommender systems are currently known to be the most popular and successful approach in recommender systems and are widely used in e-commerce websites [14]. By finding neighbors similar to a user's profile, collaborative filtering algorithms provide taste-based recommendations of neighbors that are thought to represent different people's interests. In most of these websites, anyone can submit and post their opinion about a specific item. Interactivity of the collaborative filtering on the one hand has created power and on the other hand has caused the vulnerability of this type of recommender system [15].

These issues have left collaborative filtering-based recommender systems vulnerable to various types of shilling attacks by profit-seeking people, which are one of the most common attacks in these systems. These attacks usually come into two forms in recommender systems. In the first case, the beneficiary posts positive feedback in favor of the desired product, and in the second case, the beneficiary posts negative feedback against the product or competing products [16, 17]. Posting fake feedback in recommender systems can alter the results and reduce the accuracy of the system's recommendations. Therefore, it seems essential to detect shilling attacks and neutralize their effects in recommender systems.

several methods and algorithms have been developed in this regard. Some of these methods either examine the rating matrix from a single point of view or use low-order interactions or high-order interactions. The rating patterns of fake users and normal users become similar when an attacker uses obfuscation techniques. Shilling attacks cannot be detected by methods that only examine a single monitor's user's rating matrix. However, Schilling attack detection methods based on another unitary view of only low-order interactions or high-order interactions also suffer from low accuracy. This research provides a mechanism based on social network analysis for better detection and a lower error percentage to detect shilling attacks and better results.

Injection of fake profiles for shilling attack by the adversary with certain strategies and patterns are injected into the system. Therefore, there are certain relationships between the characteristics of fake profiles, the rating time by fake profiles, and the rating matrix, which can greatly distinguish these profiles from normal profiles. In this article, the social network between profiles was drawn to find hidden patterns between fake profiles. Since fake profiles are created in almost identical ways, connections between fake profiles are denser than real profiles. Then, social network output information, rating time, and users' rating matrix were used to discover low-order and high-order information simultaneously.

Finally, a powerful tool is needed to detect and predict fake profiles and normal profiles from low-order and high-order information. Gaussian-Rough neural network was used in this study because neural networks are powerful modeling tools with unique properties and can solve nonlinear and complex problems, pattern classification, pattern recognition, and prediction. Gaussian-Rough neural networks can classify complex patterns and remove noise. In addition, the emotional learning method was used in the neural network training process. This method can properly affect network learning by considering the errors of the previous moments.

The basic concepts are introduced in section 2. In section 3, the background is reviewed. In section 4, the proposed mechanism is presented, and then in section 5, an experimental evaluation is done to check the results. Finally, in section 6 the conclusion is presented.

2. BASIC CONCEPTS

In this section, we will introduce and briefly explain the

common definitions and terms in shilling attacks, then we will examine the model and types of shilling attacks.

2. 1. Common Definitions And Terms

- Adversary: a person or persons who intend to attack a recommender system.
- Shilling attack: an attack carried out by the adversary to post fake feedback and alter the result of the recommender system.
- Profile: a set of points given by a user to different items in the recommender system.
- Fake profiles: profiles that are injected into the system by the adversary to achieve the desired results.
- Attack intent: each type of shilling attack may have various intents, but the final intent of the adversary may be one of the following [16, 17]. The two main targets are push and nuke. In push, the adversary injects fake profiles into the system to post positive feedback to increase the probability of an item to be seen, and in nuke, the adversary injects fake profiles into the system to post negative feedback to reduce the probability of the item or competing items to be seen. Another goal of the shilling attack is random sabotage [17], which is done by disrupting recommendation algorithms to reduce users' trust in the recommender system.
- Filler size: the number of points given by the fake profile to the items in the recommender system [18]. Adding the number of points costs relatively less than creating a fake profile for the adversaries. Since normal users do not rate all the items of the recommender system, usually the filler size is between 1 and 20% of the total items.
- Attack size: the number of fake profiles injected into the recommender system by the adversary. The number of profiles injected into a recommender system is usually set between 1 and 15% [18].
- Target item: The item that the adversary intends to minimize or maximize its rating in the recommender system depending on the attack type [19].
- Low knowledge attacks: These types of attacks require little knowledge about the recommender system (such as the rating range of items).
- High knowledge attacks: These types of attacks require a high level of knowledge about the recommender system.

2. 2. Shilling Attack Model

The adversary performs the shilling attack by injecting fake profiles, which was first defined in the research by Bhaumik et al. [20] and Mobasher et al. [21] to misguide collaborative filtering-based recommender systems. Figure 1 shows the overall diagram of fake profiles in the recommender system in attacks with a single target item. But in practice, the adversary can attack several target items at the same time. Yang et al. [22] and Chung et al. [23] suggested creating attacks with several target items simultaneously. Figure 2 shows the overall diagram of the fake profiles in the recommender system in this case. In fact, the attack model can be considered as an approach to create fake profiles rely on the existing knowledge of the recommender system [20, 21].

As shown in Figures 1 and 2, the fake profiles of a recommender system in the shilling attack include an n-dimensional vector of ratings, where n represents the number of items in the system. This vector contains a set of target items i_t along with a rating function γ that assigns a rating value to it and γ_{max} rates push and γ_{min} rates nuke according to the attack intent. I_S is a set of selected items with specific characteristics determined by the adversary and typically used for group attacks. I_F is a set of filler items that are usually randomly selected along with a rating function σ to map the filler items to the rating value. The filler items are created to normalize the appearance of fake profiles and increase the difficulty of identifying them. $I_\emptyset$ is a set of items that are not rated in the fake profile. In fact, the main difference among different shilling attack models is in the selection of the set of filler items, the selected items and their rating strategies.

2. 3. Types of Shilling Attacks

According to the shilling attack model explained in the previous section, fake profiles with specific strategies and patterns are injected into the recommender system. Table 1 summarized the types of known attacks [15, 24] and their strategies against recommender systems with a collaborative refinement approach. In addition, the attack type and category, as well as I_S ،I_F ،$I_\emptyset$ ، I_t rating for

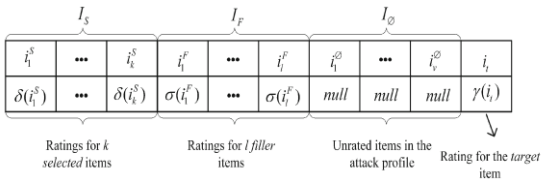

Figure 1. General diagram of fake profiles in attacks with a target item

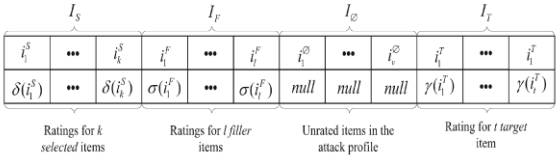

Figure 2. General diagram of fake profiles in attacks with multiple target items

famous shilling attacks are stated in Table 1. For example, I_S items are not rated in a random attack which is a basic attack category. The I_F items are randomly rated with a normal distribution around the average rating value in the entire database. $I_\emptyset$ are also not rated, and the I_F item(s) are rated according to the target of the attack, γ_{max} for push and γ_{min} for nuke.

TABLE 1. Types of shilling attacks and their strategies

Attack Model	Attack Group	Reference	I_S	I_F	$I_\emptyset$	I_T
Random attack	Basic attacks	[25, 26]	Null	Random rating with a normal distribution around the mean rating in the entire database	Null	$\gamma_{max}/\gamma_{min}$
Average attack	Basic attacks	[26]	Null	Random rating with a normal distribution around the mean rating for item i in I_F	Null	$\gamma_{max}/\gamma_{min}$
Bandwagon attack	Low-knowledge attacks	[27]	Popular items are rated with γ_{max}	Random rating with a normal distribution around the mean rating in the entire database	Null	γ_{max}
Segment attack	Low-knowledge attacks	[28]	Popular items are rated with γ_{max}	γ_{min}	It is determined based on the size of the filler item	γ_{max}
Love/hate attack	Nuke attack	[18]	Null	γ_{max}	Null	γ_{min}
Reverse bandwagon attack	Nuke attack	[21]	The least popular items are rated with γ_{min}	Random rating with a normal distribution around the mean rating for item i in I_F	Null	γ_{min}
Sampling attack	High-knowledge attacks	[29]	Null	Copy of an existing profile	It is determined based on the size of the filler item	$\gamma_{max}/\gamma_{min}$
Noise injection	Obfuscated attacks	[30]	$R_{u,i} = r_{u,i} + (\text{random number} \times \alpha)$	$R_{u,i} = r_{u,i} + (\text{random numbr} \times \alpha)$	Null	$\gamma_{max}/\gamma_{min}$
Target shifting	Obfuscated attacks	[30]	$R_{u,i} = r_{u,i}$	$R_{u,i} = r_{u,i}$	Null	$\gamma_{max} - 1 /\gamma_{min} + 1$
User shifting	Obfuscated attacks	[30]	$R_{u,i} = r_{u,i} + \text{shift}(u, O_s)$	$R_{u,i} = r_{u,i} + \text{shift}(u, O_s)$	Null	$\gamma_{max}/\gamma_{min}$
Mixed attack	Obfuscated attacks	[31]	Simultaneous injection of fake profiles of all kinds of shilling attacks			
Average over popular items (AOP) attack	Obfuscated attacks	[32]	Null	Equally likely to be selected from the top X% of most popular items.	Null	$\gamma_{max}/\gamma_{min}$
Power User Attack (PUA) attack	Other attacks	[33]	Items and ratings are copied from powerful user profiles.	Null	Null	$\gamma_{max}/\gamma_{min}$
Power Item Attack (PIA) attack	Other attacks	[34]	Powerful items are rated with a normal distribution adjusted around the item mean.	Null	Null	$\gamma_{max}/\gamma_{min}$
Bandwagon and average hybrid attack	Other attacks	[35]	Bandwagon items selected and rated with γ_{max}; Mean items are rated with a normal distribution around the item mean	Random rating with a normal distribution around the system mean	Null	$\gamma_{max}/\gamma_{min}$
Random vandalism attack	Other attacks	[17]	Random number between $[\gamma_{min}, \gamma_{max}]$	Random number between $[\gamma_{min}, \gamma_{max}]$	Null	$\gamma_{max}/\gamma_{min}$

3. BACKGROUND

Collaborative filtering-based recommender system designs are commonly developed and publicly available by e-commerce sites for customer acquisition. These systems are not sufficiently resistant to shilling attacks or fake profile injection due to their open nature [36, 37]. In general, shilling attacks cause push and nuke attacks on specific items or by injecting fake profiles to damage the performance of the recommender system.

Fraudulent behavior such as fake rating was first proposed by Dellarocas [38]. The attack on collaborative filtering-based recommender system was first introduced by O'Mahony et al. [39]. This paper defined the robustness of recommender systems and various vulnerabilities of the collaborative filtering approach against shilling attacks to promote specific recommendations.

So far, various attack detection algorithms have been presented by researchers, each of which strives to maintain the overall validity of the recommender system. In general, there are three main approaches in research including supervised learning, unsupervised learning, and semi-supervised learning.

Chirita et al. [25] presented the first shilling attack detection algorithm based on supervised classification. They introduced some factors that may be useful in analyzing the patterns of fake profiles insert for shilling attacks. They proposed two features to detect attacks: they are rating deviation from average agreement and the highest degree of similarity with neighbors. This algorithm is capable of detecting random, average and bandwagon attacks, but is unable to detect fragment and friend/hate ones.

Burke et al. [40] derived two new features based on the deviation of rates from the mean agreement. These features include the weighted deviation from the average agreement and the weighted degree of agreement. They then combined the extracted features with the KNN to do the attack detection. Williams et al [41] used machine learning algorithms including SVM to detect attacks.

Tong and Tang [42] proposed a model using interval analysis of the user ratings to detect suspicious behavior regarding the most popular items in recommender systems. They considered such features as fixed interval, frequency, and span based on the user's temporal behavior.

Xia et al. [43] presented a new dynamic interval segmentation method based on item anomaly detection to detect shilling attacks. Yang et al. [44] proposed three new features that focus on a number of specific rates (such as maximum rate, minimum rate, and average rate) in the selected items to deal with the imbalanced classification problem. This method attempts to identify all fake profiles from the real ones.

Using classic machine learning algorithms, Wu et al. [45] selected two attack detection methods based on highest performance features. Li et al. [46] used a statistical analysis method based on item popularity. This method compares and examines the popularity distribution among attack and normal profiles.

As with the semi-supervised learning methods, Wu et al. [47] presented a hybrid shilling attack detector to detect more complex shilling attacks. First, this algorithm collects the criteria of well-known shilling attacks in order to select the feature through an overlay. The algorithm then uses simple semi-supervised Bayes classification to group labeled and unlabeled users.

In the unsupervised learning approach, Mehta [48] proposed a method called PCASelectUsers. To identify fake profiles, this method requires obtaining certain information. Yang et al. [22] proposed a method based on graph mining. In this method, they used a clustering algorithm to calculate the similarity of normal users and suspicious users.

Shao and Sun [49] proposed a method named XGB-SAD to detect the shilling attack by binary combination of gradient boosting schematics. They analyzed the rating matrix with a binary schematic of time and item with using eXtreme Gradient Boosting.

The methods mentioned above either examine the rating matrix from a single point of view or use low-order interactions or high-order interactions. In this article, social network output information (in order to find hidden patterns), users' rating time and rating matrix were simultaneously used to use low-order and high-order information in discovering shilling attacks. The proposed mechanism and its details are explained in the next sections.

4. PROPOSED MECHANISM

This section presents the details of the proposed mechanism in five stages, including the Injection of Shilling Attacks, Creating a Social Network of Users, Neural Network Inputs, Building a Gaussian Rough Neural Network With Emotional Learning and Detection of Fake Profiles according to Figure 3. Table 2 shows the actions and objectives of the above steps.

4. 1. Injection of Shilling Attacks The first step of the proposed method is to inject the shilling attack(s) into the recommender system. Since there is no data set containing types of shilling attacks of different attack sizes, this issue can lead to wrong injection of shilling attacks into the system. If shilling attacks are mistakenly injected into the system, they can interfere with the evaluation of the proposed method and make it difficult to recognize fake profiles and detect shilling attacks.

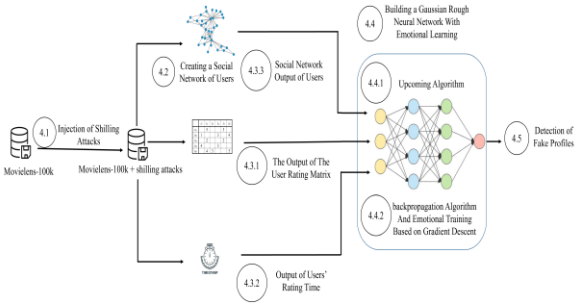

Figure 3. General diagram of the proposed mechanism

Therefore, the first stage of the proposed method is particularly important, and the implementation of the following stages depends on the precise implementation of this stage.

At this stage, fake profiles were injected into the data set using the shilling attack model, and the resulting rating matrix including fake and normal profiles was used as input for the next step.

4. 2. Creating a Social Network of Users A social network is a social structure that consists of a number of social actors and there are binary relationships (social relationships) among these social actors. Social actors are not necessarily human and a group of any gender may form a community. For example, a group of humans, a group of buffaloes, a group of computers and a group of robots are examples of community. Social relations may also exist in various forms between social actors. For example, friendship, interest, trust, cooperation, etc. are considered as social relationships. One of the main goals of social network mapping is to study collective behavior.

In other words, there are patterns in the structure of social networks, by using these patterns we are able to discover knowledge from the network and predict the future of the network. A series of recent discoveries show us the amazing truth that a number of simple and inaccessible rules govern the structure and evolution of social networks, although these rules are very complex until they are not known.

One of the best ways to model social networks is to use graph theory because social actors can be imagined as vertices and social relationships between them as edges [50]. In fact, the starting point of social networks can be traced back to 1735 with the emergence of graph theory.

In this step, a social network is drawn using user profiles, then based on this social network, information is sent to the neural network as an input. At first, according to the profile of users represented by P, which is the output of stage 1 and includes real and fake profiles, an undirected user-user network G = (V, E, W) is formed, where V is a set of vertices and E represents the set of edges between vertices, W is a weight matrix where each element wij ∈ W shows the weight corresponding to the edge eij. For example, Iu and Iv are two item vectors

TABLE 2. Actions and objectives of the proposed mechanism

Stage		Action	Objective
Injection of shilling attacks		Injects fake profiles into the data set using the shilling attack model	Creating a dataset including fake and normal profiles
Creating a social network of users		Creating a social network using relationships between items and profiles	Discover knowledge from the network and finding hidden patterns between profiles
Neural network inputs	The output of the user rating matrix	Calculate Boolean values of user rating, The coefficient of item Boolean, Mean index of user Boolean, The number of max and min rating and Max-1 and min+1 rating number	Reducing the negative effects of obfuscation techniques and using low-order interactions to detecting shilling attacks
	Output of users' rating time	Calculate Collection of user rating time, The max interval of user rating time, Aggregation index of user rating time and Relative aggregation index of user rating time	Using rating intervals to detecting shilling attacks
	Social network output of users	Calculate Degree centrality, Closeness centrality, Eigenvector centrality and Local clustering coefficient	Using high-order interactions to detecting shilling attacks
Building a gaussian rough neural network with emotional learning	Upcoming algorithm Backpropagation algorithm and emotional training based on gradient descent	Training weights, cluster centers, and standard deviation	Reducing the error between the network output and the desired output
detection of fake profiles		Using the proposed model with test data	Separation normal and fake profiles

rated by users u and v, respectively. An edge is created between vertices u and v if $| I_u \cap I_v | > t$, where t is an empirical threshold. Additionally, the weight of each edge is set to 1 due to its undirected graph (as described in Figure 4).

Since fake profiles are created in almost identical ways, it means that the communication between attackers is denser than that of real users. During the process of building a social network, it is very important how to choose the threshold t to detect all attackers and filter out more real users at the same time. Yang et al.'s paper [22] has been used to determine how to choose the threshold t.

4. 3. Neural Network Inputs According to Figure 3, the inputs of the neural network are provided by the outputs of users' rating matrix, users' rating time and users' social network. At this stage, these items will be reviewed.

4. 3. 1. The Output of The User Rating Matrix The attacker can design attacks by using the knowledge gained from the recommender system and obfuscation methods and insert profiles into the system that are similar to the existing normal profiles. Therefore, attack detection methods that use low-order features (user rating matrix) make mistakes in evaluating normal and fake users. To reduce the effects of this issue, in addition to analyze the rating matrix, Boolean values of user ratings are also considered.

- Boolean values of user rating

In this method, instead of considering the value of user rating to an item, only the user's rating to an item is considered. So we make the user's rating values (R_{ij} reprsents the rating value of $user_i$to $item_j$ in the rating matrix $R_{m\times n}$).

```
Input:
    User Profile P;
    Empirical threshold t;

Out Put:
    Undirected Social Network G;

Process:
    for each pair of users u, v ∈ P do;
      if | I u ∩ I v | > t then
        Create an edge e u v between vertexes u and v , set w u v = 1 ;
      end if
    end for

Construct an undirected Social Network G;
return G ;
```

Figure 4. Algorithm of social network construction

$$\mathrm{BVUR_{ij}} = \begin{cases} 0\,, & \mathrm{R_{i,j}} = 0 \\ 1\,, & \mathrm{R_{i,j}} \neq 0 \end{cases} \quad (1)$$

- The coefficient of item Boolean

Coefficient of item Boolean is equal to the sum of BVUR values of all users in a column. This coefficient shows the number of times each item is rated as well as the acceptance rate of the item. TCIB value for item j is defined as follows:

$$\mathrm{TCIB_j} = \sum_{i=1}^{n} \mathrm{BVUR_{ij}} \quad (2)$$

- Mean index of user Boolean

First, the product of the coefficient of item Boolean and the Boolean value of the user's rating to the items are added from the beginning to the end, respectively. Then the accumulated value is divided by the total number of users (m refers to the total number of users and n refers to the total number of items). This index is used to reduce the negative effects of obfuscation techniques.

$$MIUB_{user_i} = \frac{1}{m}\sum_{j=1}^{n} TCIB_j \times BVUR_{i,j} \quad (3)$$

- The number of max and min rating

In order to achieve their attack goals, attackers attack one or more target items with the lowest or highest ratings, which means that if the attackers want to downgrade or upgrade the items in the recommendation list, will focus on these items frequently [51-53]. The number of maximum and minimum ratings of the user is also sent to the neural network as a parameter.

- Max-1 and min+1 rating number

In some attacks, attackers may attack one or more target items with min+1 or max-1 ratings with target change attacks. The number of max-1 and min+1 rates of the user is also sent to the neural network as a parameter.

4. 3. 2. Output of Users' Rating Time A shilling attack by an attacker occurs by inserting fake profiles in a certain time unit. A shilling attack on recommender systems is a short-range action which is highly evident in the rating intervals of fake profiles. The rating interval for fake profiles is significantly different from normal profiles [54, 55]. Based on this, the following items are extracted from users' rating time according to the following equations:

- Collection of user rating time

In this collection, there are user rating time tags for items, which are arranged in descending order (u refers to a specific user and n refers to n items rated by this user).

$$\mathrm{CURT_u} = \{\mathrm{t_1}, \mathrm{t_2}, \mathrm{t_3}, \ldots, \mathrm{t_n}\} \quad (4)$$

- The max interval of user rating time

$$\mathrm{MIURT_u} = \mathrm{CURT_n} - \mathrm{CURT_1} \quad (5)$$

- Aggregation index of user rating time

$$AIURT_u = \frac{MIURT_u}{N_u} \quad (6)$$

- Relative aggregation index of user rating time

$\overline{MIURT}$ is the mean MIUTR values of the users in the database and $\overline{N}$ is the mean of all user-rated items.

$$RAIURT_u = \frac{|MIURT_u - \overline{MIURT}|}{|N_u - \overline{N}|} \quad (7)$$

4. 3.3. Social Network Output of Users

- Degree centrality

This measure calculates the number of neighbors of a vortex. In fact, this index determines the degree of connection of a vortex with other vortices, which expresses the social connections of a vortex. This measure is calculated by dividing the degree of each vortex k_i by N-1, where N is the number of vortices in the entire network [56].

$$C_D(i) = \frac{k_i}{N-1} \quad (8)$$

- Closeness centrality

A vortex is located in the center of a network when it can quickly interact with other vortices. This measure calculates the average length of the shortest path from the desired vortex to other vortices of the network (d_{ij} refers to the length of the shortest path from vortex i to vortex j) [56].

$$C_C(i) = \frac{N-1}{\sum_{i\neq j}^{N} d_{ij}} \quad (9)$$

- Eigenvector centrality

Eigenvector centrality is one of the measures that shows the importance of a vortex. This index calculates the relative rating of all vortices according to a general rule. In fact, the vortex connected to high-rating vortices rates more than the vortex connected to low-rating vortex. This measure is calculated using the neighborhood matrix and according to the following equation [56]:

$$C_E(i) = \frac{1}{\lambda}\sum_j A_{i,j} C_E(i) \quad (10)$$

- Local clustering coefficient

This measure examines the relationship between the neighbors of a vortex. According to the following equation, the local clustering coefficient for vortex i is calculated as the result of dividing the number of links between friends of vortex i by the number of possible edges between friends of vortex i [56].

$$C_i = \frac{2e_i}{k_i(k_i-1)} \quad (11)$$

4. 4. Building a Gaussian Rough Neural Network With Emotional Learning

Neural network is a branch of computational intelligence that tries to solve problems based on abstract structure. The performance of neural networks is based on training and information sampling. The important factor in neural networks are neuronal units. Although neurons are a simple computational transformation function, the network structure by combining these neurons can be used in simple and complex systems that can solve small and large problems. As a result, neural networks are able to solve problems with different behavior and dynamics. Neural networks are widely used with the aim of human-like performance these days. These networks are composed of a number of non-linear computing elements that operate in parallel [57, 58].

At this stage, Gaussian rough neural network has been used to classify profiles and detect shilling attacks. Gaussian neural networks are usually used in function approximation, interpolation and classification. In general, the method that the RBF neural network uses to classify complex patterns is based on a non-linear mapping from the n_0 dimensional space (the number of input parameters) to the larger m dimensional space (the number of intermediate layer neurons). According to Cover's theorem, after a nonlinear mapping to a higher dimensional space, complex patterns can be linearly classified better than the initial space with lower dimensions. On the other hand, real data is always associated with uncertainty and neural networks do not perform well in the presence of noisy data. One of the noise-resistant neural networks is the rough neural network.

According to Figure 5, a Gaussian rough neural network has been used to classify normal and fake profiles. If we consider the input vector as follows:

$$x = [x_1, x_2, \dots, x_{n0}] \quad (12)$$

4. 4. 1. Upcoming Algorithm

$$\|x - c_j\| = \sqrt{(x_1 - c_1^1)^2 + \cdots + (x_{n0} - c_{n0}^1)^2} \quad (13)$$

$$o_j^1 = \varphi_j(net_j^1) = \exp\left[-\frac{1}{2}\left[\frac{\|x-c_j\|}{\sigma_j}\right]^2\right] \quad (14)$$

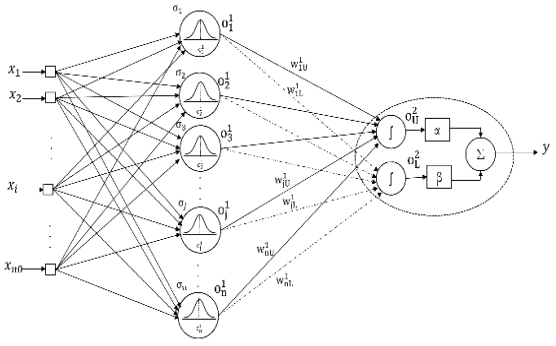

Figure 5. Gaussian rough neural network design for problem solving

$$o_j^1 = \exp\left[-\frac{1}{2(\sigma_j)^2}\sum_{p=1}^{n_0}(x_p - c_{pj})^2\right] \quad (15)$$

The output of Gaussian rough network for the upper and lower limit is as:

$$o_U^2 = \max\left[w_U^T(k)o^1(k), w_L^T(k)o^1(k)\right] \quad (16)$$

$$o_L^2 = \min\left[w_U^T(k)o^1(k), w_L^T(k)o^1(k)\right] \quad (17)$$

And the output of the network is finally as:

$$y(k) = \propto o_L^2 + \beta o_U^2 \quad (18)$$

4. 4. 2. Backpropagation Algorithm And Emotional Training Based on Gradient Descent In this section, using the error between the network output and the desired output, neural network parameters, including weights, cluster centers, and standard deviation, are taught. For better learn these parameters, emotional training algorithm and gradient descent have been used. We define the total error relation as follows:

$$E(k) = \frac{1}{2}\sum_{i=1}^{N}(r)^2(k) = \frac{1}{2}\sum_{i=1}^{N}(k_1 e_i(k) + k_2 \dot{e}_i(k))^2 \quad (19)$$

$$E(k) = \frac{1}{2}\sum_{i=1}^{N}((k_1 + k_2)e_i(k) - k_2 e_i(k-1))^2 \quad (20)$$

First mode if $w_U^T(k)o^1(k) \geq w_L^T(k)o^1(k)$:

$$w_U(k+1) = w_U(k) + \eta_w(k_1 + k_2)r(k) \propto o_j^1(k) \quad (21)$$

$$w_L(k+1) = w_L(k) + \eta_w(k_1 + k_2)r(k)\beta o_j^1(k) \quad (22)$$

$$c_j(k+1) = c_j(k) + \eta_c(k_1 + k_2)r(k) \left[\propto w_{Uj}(k) + \beta w_{Lj}(k)\right]\frac{(x - c_j(k))}{(\sigma_j(k))^2}o_j^1(k) \quad (23)$$

$$\sigma_j(k+1) = \sigma_j(k) + \eta_\sigma(k_1 + k_2)r(k) [\propto w_{Uj}(k) + \beta w_{Lj}(k)]\frac{(x - c_j(k))}{(\sigma_j(k))^2}o_j^1(k) \quad (24)$$

Second mode if $(k)o^1(k) < w_L^T(k)o^1(k)$:

$$w_U(k+1) = w_U(k) + \eta_w(k_1 + k_2)r(k)\beta o_j^1(k) \quad (25)$$

$$w_L(k+1) = w_L(k) + \eta_w(k_1 + k_2)r(k)\alpha o_j^1(k) \quad (26)$$

$$c_j(k+1) = c_j(k) + \eta_c(k_1 + k_2)r(k) [\beta w_{Uj}(k) + \alpha w_{Lj}(k)]\frac{(x - c_j(k))}{(\sigma_j(k))^2}o_j^1(k) \quad (27)$$

$$\sigma_j(k+1) = \sigma_j(k) + \eta_\sigma(k_1 + k_2)r(k) [\beta w_{Uj}(k) + \alpha w_{Lj}(k)]\frac{(x - c_j(k))}{(\sigma_j(k))^2}o_j^1(k) \quad (28)$$

4. 5. Detection of Fake Profiles The process of detecting shilling attacks in the proposed model is done in four stages. In the first step, fake profiles are inserted into the system using the shilling attack model and attack parameters. The resulting rating matrix, after injecting shilling attacks, is used as input for the next steps.

Then, in the second stage, the social network of users is created to find patterns between users and discover knowledge. The purpose of creating a social network of users is to discover latent relationships between profiles in the network.

In the third stage, parameters are extracted from the users' social network, the users' rating matrix and the users' rating time and are used as input for the next stage.

Finally, in the fourth stage, the construction of Gaussian rough neural network is done by determining forward and back error propagation algorithms, determining the parameters of the neural network, such as determining the number of neurons, training rate, initializing the weights and biases, the number of IPACs, and determining the volume of training data, evaluation and testing. After learning the network with training data and selecting the best trained weights, the network output is checked with test data to evaluate the performance of the proposed model.

5. EXPERIMENTAL EVALUATION

5. 1. Preliminaries In this section, we will discuss the pre-test preparations containing the data set used, attack size, filler size, attack model and comparison algorithms. Movielens-100k dataset is used in this research [59]. The Movielens-100k dataset includes rating information for 1682 items from 943 users. Table 3 summarized the user-item rating table for the Movielens-100k dataset. In the rating matrix, the user's rating values for the items are from 1 to 5. 0 indicates no rate, 1 indicates the lowest rate, and 5 indicates the highest user rate for an item.

The parameters of the attack size and the filler size should be determined during shilling attack injection. The attack size parameter indicates how many fake profiles are injected into the system and the filler size

TABLE 3. User-item rating table for Movielens-100k dataset

User /item	$item_1$	$item_2$	$item_3$	$item_4$	…	$item_{1682}$
$user_1$	5	3	4	3	…	0
$user_2$	4	0	0	0	…	0
$user_3$	0	0	0	0	…	0
$user_4$	0	0	0	0	…	0
…	…	…	…	…	…	0
$user_{943}$	0	5	0	0	…	0

indicates the number of items rated by fake profiles. Attack size and padding size are defined as follows.

The attack size equal to the number of fake profiles injected into the system refers to the total number of profiles in the system database and is calculated as follows:

$$\text{Attack Size} = \frac{N_{\text{fake profiles}}}{N_u} \quad (29)$$

The filler size equal to the number of points given by fake profiles injected into the system to the items in the recommender system refers to the total number of items in the system database and is calculated as follows:

$$\text{Filler Size} = \frac{N_{IF}}{N_{item}} \quad (30)$$

This mechanism is compared with four methods used in the experiments: PCA [48], Semi [60] and BAY [61] and XGB [49] to compare the performance of the proposed mechanism.

PCA is a method that uses unsupervised learning method PCA-SelectUsers to identify malicious fake users. Semi is a semi-supervised learning method. BAY combines several sets of base classifiers and uses the combined output to detect the shilling attack. XGB is a method that utilizes binary combination of gradient boosting to detect shilling attacks. Also, average attack, random attack and bandwagon attack models are used in this research.

5. 2. Evaluation Criteria In this research, three efficiency measures of shilling attack detection schemes are used. These criteria are:

- Precision

expressed as the percentage of fake profiles actually detected divided by all fake profiles [20].

$$\text{Precision} = \frac{TP}{TP+FP} \quad (31)$$

- Recall

expressed as the percentage of fake profiles actually detected divided by all fake profiles [20].

$$\text{Recall} = \frac{TP}{TP+FN} \quad (32)$$

- F1-Measure

combines precision and recall rate [20].

$$F1 - \text{Measure} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (33)$$

5. 3. Evaluation Results In this section, we test three models of average attack, random attack and bandwagon attack with parameters of 10% filler size and 3, 5, 7, 10, 12 and 15% attack size after neural network learning. Figures 6, 7, and 8 show the performance of the proposed mechanism for precision, recall, and F1 criteria, respectively.

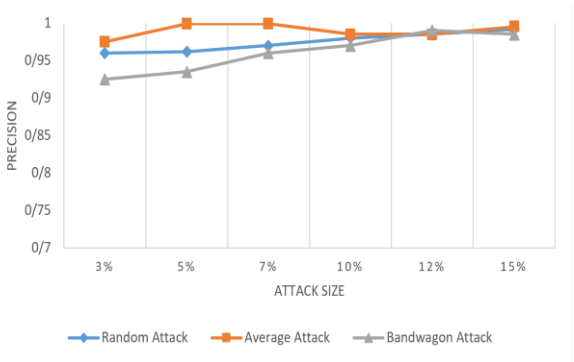

Figure 6. Performance of the proposed mechanism for precision

Figure 6 shows a view of the changes in the precision. As it is clear from the graph, in general, the precision of the proposed mechanism is improving as the attack size increases. Also, the precision in small and large attacks is above 0.9 and is at an acceptable level.

Figure 7 shows a view of the recall changes. According to the figure, the proposed mechanism in the random attack model, compared to the average and bandwagon attack, works weaker in small-sized attacks, but with an increase in the size of the attack, the recall in the proposed mechanism is generally increased.

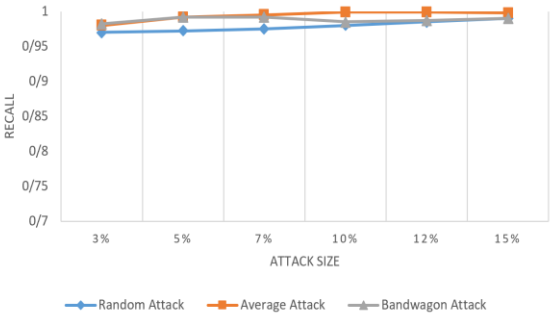

Figure 7. Performance of the proposed mechanism for recall

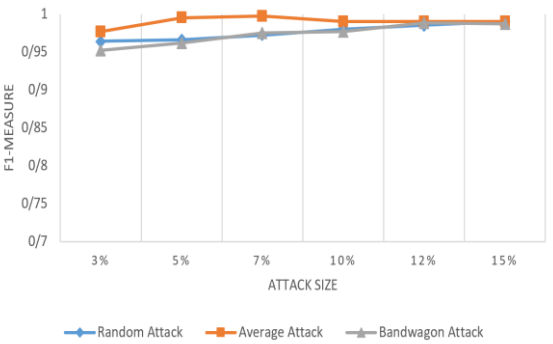

Figure 8. Performance of the proposed mechanism for F1-Measure

Figure 8 shows a view of the changes in the F1. F1 is generally improving as the attack size increases and is almost above 0.95 and is at an acceptable level. based on the experiments the comprehensive detection effectiveness of the algorithm is better in average attacks.

After checking the results of the proposed mechanism, the performance of the proposed mechanism is compared with the four methods used in PCA, Semi, BAY and XGB experiments. For this purpose, three models of random, average and bandwagon attacks were tested with parameters like 10% filler size and 3, 5, 7, 10, 12 and 15% attack size for the mentioned methods. Figures 9, 10, and 11 show the performance of the proposed mechanism for random attack, average attack, and bandwagon attack model, respectively, for F1.

As can be seen in Figures 9, 10 and 11, the proposed mechanism is clearly more effective in detecting attacks in random, average and bandwagon attack models.

6. DISCUSSION AND CONCLUSION

A growing number of e-commerce sites are implementing recommender systems to solve the selection overhead problem. The open and interactive nature of recommender systems has made it possible for adversaries to disrupt their proper functioning by recording fake feedback through shilling attacks. Thus, the early detection of hose attacks in recommender systems plays a very important role in maintaining the stability of the recommender system and, along with it, maintaining its credibility.

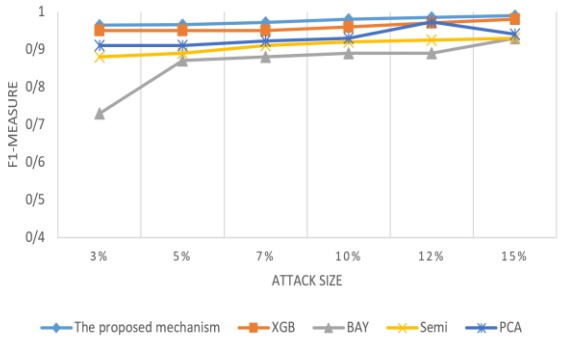

Figure 9. Performance of comparison methods under F1-Measure evaluation criterion for random attack

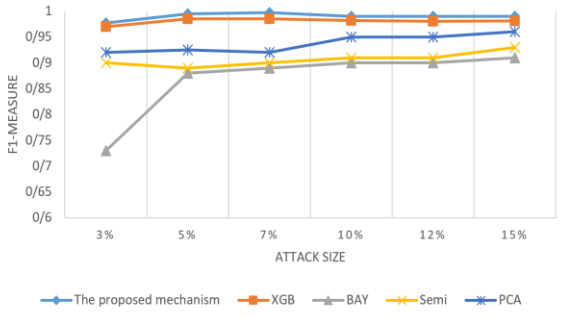

Figure 10. Performance of comparison methods under F1-Measure evaluation criterion for average attack

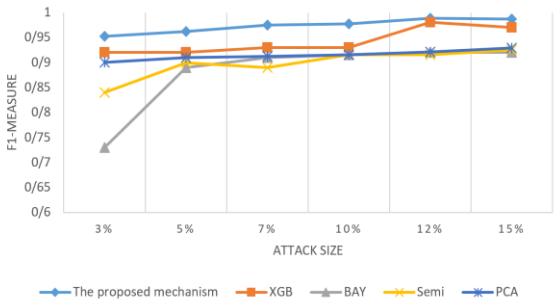

Figure 11. Performance of comparison methods under F1-Measure evaluation criterion for bandwagon attack

This paper presented a new mechanism for detecting shilling attacks using social network analysis and Gaussian-Rough neural network. Fake profiles with specific strategies and patterns are injected into the recommender system, and identifying the characteristics of these strategies and patterns detects shilling attacks and discover fake profiles. The three outputs of users' rating matrix, rating time, and analysis of users' social networks were used to discover low and high order information after modeling their profiles and features in the form of a network of vertices, and edges, and building a social network at the same time. This type of neural network was used to detect fake profiles due to the high ability of Gaussian-Rough neural networks to classify complex patterns and noise resistance.

The proposed mechanism overcomes the limitations of previous methods and analyzes user profiles from different perspectives, as well as uses low-order interactions and high-order interactions to detect malicious attackers. The experimental results show that the proposed mechanism in the mean and bandwagon random attack model is more effective in detecting attacks compared to the four methods PCA, Semi, BAY, and XGB.

The proposed mechanism can be used as a practical method in recommender systems based on collaborative filtering in e-commerce sites to detect standard attacks. The main challenge facing the proposed mechanism is group shilling attacks. The proposed mechanism for detecting group shilling attacks is considered in future research because the shilling attack detection algorithms mainly focus on identifying individual attackers in online recommender systems and rarely deal with group shilling attacks.

7. REFERENCES

1. Bobadilla, J., Ortega, F., Hernando, A. and Gutiérrez, A., "Recommender systems survey", ***Knowledge-based systems***, Vol. 46, (2013), 109-132. doi: 10.1016/j.knosys.2013.03.012.
2. Resnick, P. and Varian, H.R., "Recommender systems", ***Communications of the ACM***, Vol. 40, No. 3, (1997), 56-58. doi: 10.1145/245108.245121.
3. Adomavicius, G. and Tuzhilin, A., "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions", ***IEEE transactions on knowledge and data engineering***, Vol. 17, No. 6, (2005), 734-749. doi: 10.1109/TKDE.2005.
4. Aggarwal, C.C., "Recommender systems, Springer, Vol. 1, (2016).
5. Moghaddam, F.B. and Elahi, M., Cold start solutions for recommendation systems, in Big data recommender systems: Recent trends and advances. 2019, IET.
6. Bollen, D., Knijnenburg, B.P., Willemsen, M.C. and Graus, M., "Understanding choice overload in recommender systems", in Proceedings of the fourth ACM conference on Recommender systems., (2010), 63-70.
7. Ricci, F., Rokach, L. and Shapira, B., Introduction to recommender systems handbook, in Recommender systems handbook. 2011, Springer.1-35.
8. Jannach, D., Zanker, M., Felfernig, A. and Friedrich, G., "Recommender systems: An introduction, Cambridge University Press, (2010).
9. Rubens, N., Elahi, M., Sugiyama, M. and Kaplan, D., Active learning in recommender systems, in Recommender systems handbook. 2015, Springer.809-846.
10. Su, X. and Khoshgoftaar, T.M., "A survey of collaborative filtering techniques", ***Advances in Artificial Intelligence***, Vol. 2009, (2009). https://doi.org/10.1155/2009/421425
11. Burke, R., "Hybrid recommender systems: Survey and experiments", ***User Modeling and User-adapted Interaction***, Vol. 12, No. 4, (2002), 331-370.
12. Alonso, S., Bobadilla, J., Ortega, F. and Moya, R., "Robust model-based reliability approach to tackle shilling attacks in collaborative filtering recommender systems", ***IEEE Access***, Vol. 7, (2019), 41782-41798. doi: 10.1109/ACCESS.2019.2905862.
13. Kaminskas, M. and Bridge, D., "Diversity, serendipity, novelty, and coverage: A survey and empirical analysis of beyond-accuracy objectives in recommender systems", ***ACM Transactions on Interactive Intelligent Systems (TiiS)***, Vol. 7, No. 1, (2016), 1-42. doi: 10.1145/2926720.
14. Jia, C.-X. and Liu, R.-R., "Improve the algorithmic performance of collaborative filtering by using the interevent time distribution of human behaviors", ***Physica A: Statistical Mechanics and its Applications***, Vol. 436, (2015), 236-245. doi: 10.1016/j.physa.2015.05.060.
15. Si, M. and Li, Q., "Shilling attacks against collaborative recommender systems: A review", ***Artificial Intelligence Review***, Vol. 53, No. 1, (2020), 291-319. doi: 10.1007/s10462-018-9655-x.
16. Mobasher, B., Burke, R., Bhaumik, R. and Williams, C., "Toward trustworthy recommender systems: An analysis of attack models and algorithm robustness", ***ACM Transactions on Internet Technology (TOIT)***, Vol. 7, No. 4, (2007), 23-es. doi: 10.1145/1278366.1278372.
17. Burke, R., O'Mahony, M.P. and Hurley, N.J., Robust collaborative recommendation, in Recommender systems handbook. 2015, Springer.961-995.
18. Williams, C.A., "Thesis: Profile injection attack detection for securing collaborative recommender systems", (2012).
19. O'Mahony, M.P., Hurley, N.J. and Silvestre, G.C., "Recommender systems: Attack types and strategies", in AAAI., (2005), 334-339.
20. Bhaumik, R., Williams, C., Mobasher, B. and Burke, R., "Securing collaborative filtering against malicious attacks through anomaly detection", in Proceedings of the 4th workshop on intelligent techniques for web personalization (ITWP'06), Boston. Vol. 6, (2006), 10.
21. Mobasher, B., Burke, R., Bhaumik, R. and Sandvig, J.J., "Attacks and remedies in collaborative recommendation", ***IEEE Intelligent Systems***, Vol. 22, No. 3, (2007), 56-63. doi: 10.1109/MIS.2007.45.
22. Yang, Z., Cai, Z. and Guan, X., "Estimating user behavior toward detecting anomalous ratings in rating systems", ***Knowledge-based Systems***, Vol. 111, (2016), 144-158. doi: 10.1016/j.knosys.2016.08.011.
23. Chung, C.-Y., Hsu, P.-Y. and Huang, S.-H., "Bp: A novel approach to filter out malicious rating profiles from recommender systems", ***Decision Support Systems***, Vol. 55, No. 1, (2013), 314-325. doi: 10.1016/j.dss.2013.01.020.
24. Rezaimehr, F. and Dadkhah, C., "A survey of attack detection approaches in collaborative filtering recommender systems", ***Artificial Intelligence Review***, Vol. 54, No. 3, (2021), 2011-2066. doi: 10.1007/s10462-020-09898-3.
25. Chirita, P.-A., Nejdl, W. and Zamfir, C., "Preventing shilling attacks in online recommender systems", in Proceedings of the 7th annual ACM international workshop on Web information and data management., (2005), 67-74.
26. Lam, S.K. and Riedl, J., "Shilling recommender systems for fun and profit", in Proceedings of the 13th international conference on World Wide Web., (2004), 393-402.
27. O'Mahony, M.P., Hurley, N.J. and Silvestre, G.C., "Detecting noise in recommender system databases", in Proceedings of the 11th international conference on Intelligent user interfaces., (2006), 109-115.
28. Mobasher, B., Burke, R., Bhaumik, R. and Williams, C., "Effective attack models for shilling item-based collaborative filtering systems", in Proceedings of the WebKDD Workshop, Citeseer., (2005), 13-23.
29. Burke, R., Mobasher, B. and Bhaumik, R., "Limited knowledge shilling attacks in collaborative filtering systems", in Proceedings of 3rd international workshop on intelligent techniques for web personalization (ITWP 2005), 19th international joint conference on artificial intelligence (IJCAI 2005)., (2005), 17-24.
30. Williams, C., Mobasher, B., Burke, R., Sandvig, J. and Bhaumik, R., "Detection of obfuscated attacks in collaborative recommender systems", in Proceedings of the ECAI'06 Workshop on Recommender Systems. Vol. 94, (2006).
31. Bhaumik, R., Mobasher, B. and Burke, R., "A clustering approach to unsupervised attack detection in collaborative recommender systems", in Proceedings of the International Conference on Data Science (ICDATA), Citeseer. (2011), 1.
32. Hurley, N., Cheng, Z. and Zhang, M., "Statistical attack detection", in Proceedings of the third ACM conference on Recommender systems., (2009), 149-156.
33. Wilson, D.C. and Seminario, C.E., "When power users attack: Assessing impacts in collaborative recommender systems", in Proceedings of the 7th ACM conference on Recommender systems., (2013), 427-430.
34. Seminario, C.E. and Wilson, D.C., "Nuking item-based collaborative recommenders with power items and multiple targets", in The Twenty-Ninth International Flairs Conference. (2016).

35. Zhang, F., "Analysis of bandwagon and average hybrid attack model against trust-based recommender systems", in 2011 Fifth International Conference on Management of e-Commerce and e-Government, IEEE., (2011), 269-273.
36. O'Mahony, M., Hurley, N., Kushmerick, N. and Silvestre, G., "Collaborative recommendation: A robustness analysis", ***ACM Transactions on Internet Technology (TOIT)***, Vol. 4, No. 4, (2004), 344-377. doi: 10.1145/1031114.1031116.
37. Mobasher, B., Burke, R. and Sandvig, J.J., "Model-based collaborative filtering as a defense against profile injection attacks", in AAAI. Vol. 6, (2006), 1388.
38. Dellarocas, C., "Immunizing online reputation reporting systems against unfair ratings and discriminatory behavior", in Proceedings of the 2nd ACM Conference on Electronic Commerce., (2000), 150-157.
39. O'Mahony, M.P., Hurley, N.J. and Silvestre, G., "Promoting recommendations: An attack on collaborative filtering", in International Conference on Database and Expert Systems Applications, Springer., (2002), 494-503.
40. Burke, R., Mobasher, B., Williams, C. and Bhaumik, R., "Classification features for attack detection in collaborative recommender systems", in Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining., (2006), 542-547.
41. Williams, C.A., Mobasher, B. and Burke, R., "Defending recommender systems: Detection of profile injection attacks", ***Service Oriented Computing and Applications***, Vol. 1, No. 3, (2007), 157-170. doi: 10.1007/s11761-007-0013-0.
42. Tang, T. and Tang, Y., "An effective recommender attack detection method based on time sfm factors", in 2011 IEEE 3rd International Conference on Communication Software and Networks, IEEE., (2011), 78-81.
43. Xia, H., Fang, B., Gao, M., Ma, H., Tang, Y. and Wen, J., "A novel item anomaly detection approach against shilling attacks in collaborative recommendation systems using the dynamic time interval segmentation technique", ***Information Sciences***, Vol. 306, (2015), 150-165. doi: 10.1016/j.ins.2015.02.019.
44. Yang, Z., Xu, L., Cai, Z. and Xu, Z., "Re-scale adaboost for attack detection in collaborative filtering recommender systems", ***Knowledge-based Systems***, Vol. 100, (2016), 74-88. doi: 10.1016/j.knosys.2016.02.008.
45. Wu, Z.-A., Zhuang, Y., Wang, Y.-Q. and Cao, J., "Shilling attack detection based on feature selection for recommendation systems", ***Acta Electonica Sinica***, Vol. 40, No. 8, (2012), 1687. doi: 10.3969/j.issn.0372-2112.2012.08.031.
46. Li, W., Gao, M., Li, H., Zeng, J., Xiong, Q. and Hirokawa, S., "Shilling attack detection in recommender systems via selecting patterns analysis", ***IEICE Transactions on Information and Systems***, Vol. 99, No. 10, (2016), 2600-2611. doi: 10.1587/transinf.2015EDP7500.
47. Wu, Z., Wu, J., Cao, J. and Tao, D., "Hysad: A semi-supervised hybrid shilling attack detector for trustworthy product recommendation", in Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining., (2012), 985-993.
48. Mehta, B., "Unsupervised shilling detection for collaborative filtering", in AAAI., (2007), 1402-1407.
49. Shao, C. and zhong yi Sun, Y., "Shilling attack detection for collaborative recommender systems: A gradient boosting method", ***Mathematical Biosciences and Engineering***, Vol. 19, No. 7, (2022), 7248-7271. doi: 10.3934/mbe.2022342.
50. Ajzen, I., "The theory of planned behavior", ***Organizational Behavior and Human Decision Processes***, Vol. 50, No. 2, (1991), 179-211. doi: 10.1016/0749-5978(91)90020-T.
51. Zhou, W., Koh, Y.S., Wen, J., Alam, S. and Dobbie, G., "Detection of abnormal profiles on group attacks in recommender systems", in Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval., (2014), 955-958.
52. Zhou, W., Wen, J., Koh, Y.S., Alam, S. and Dobbie, G., "Attack detection in recommender systems based on target item analysis", in 2014 International Joint Conference on Neural Networks (IJCNN), IEEE., (2014), 332-339.
53. Zhou, W., Wen, J., Koh, Y.S., Xiong, Q., Gao, M., Dobbie, G. and Alam, S., "Shilling attacks detection in recommender systems based on target item analysis", ***PloS One***, Vol. 10, No. 7, (2015), e0130968. doi: 10.1371/journal.pone.0130968.
54. Zhang, S., Chakrabarti, A., Ford, J. and Makedon, F., "Attack detection in time series for recommender systems", in Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining., (2006), 809-814.
55. Oestreicher-Singer, G. and Sundararajan, A., "Recommendation networks and the long tail of electronic commerce", ***Mis Quarterly***, (2012), 65-83. doi: 10.2139/ssrn.1324064.
56. Kong, X., Shi, Y., Yu, S., Liu, J. and Xia, F., "Academic social networks: Modeling, analysis, mining and applications", ***Journal of Network and Computer Applications***, Vol. 132, (2019), 86-103. doi: 10.1016/j.jnca.2019.01.029.
57. Liao, H., Ding, S., Wang, M. and Ma, G., "An overview on rough neural networks", ***Neural Computing and Applications***, Vol. 27, No. 7, (2016), 1805-1816. doi: 10.1007/s00521-015-2009-6.
58. Salehi, S. and Pouyan, A., "Detecting overlapping communities in social networks using deep learning", ***International Journal of Engineering, Transactions C: Aspects*** Vol. 33, No. 3, (2020), 366-376. doi: 10.5829/IJE.2020.33.03C.01.
59. Harper, F.M. and Konstan, J.A., "The movielens datasets: History and context", ***ACM Transactions on Interactive Intelligent Systems (TiiS)***, Vol. 5, No. 4, (2015), 1-19.
60. Cao, J., Wu, Z., Mao, B. and Zhang, Y., "Shilling attack detection utilizing semi-supervised learning method for collaborative recommender system", ***World Wide Web***, Vol. 16, No. 5, (2013), 729-748. https://doi.org/10.1007/s11280-012-0164-6
61. Bhebe, W. and Kogeda, O.P., "Shilling attack detection in collaborative recommender systems using a meta learning strategy", in 2015 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC), IEEE., (2015), 56-61.

Persian Abstract

چکیده

یک سیستم توصیه‌گر بخش جدایی ناپذیر از سایت‌های تجارت الکترونیکی است. یکی از چالش‌های مهم در سیستم‌های توصیه‌گر حملات شیلینگ هستند که با استفاده از ایجاد پروفایل‌های جعلی در سیستم و امتیازدهی مغرضانه به آیتم‌ها سبب کاهش دقت و از بین رفتن عملکرد صحیح سیستم توصیه‌گر در ارائه توصیه به کاربران می‌شود. هدف اصلی حمله‌کنندگان، تغییر رتبه محتوا یا آیتم‌ها متناسب با منافعشان می‌باشد. حملات شیلینگ تهدیدی علیه اعتبار سیستم‌های توصیه‌گر است بنابراین تشخیص حملات شیلینگ در سیستم‌های توصیه‌گر برای حفظ عدالت و اعتبار آن‌ها امری ضروری به نظر می‌رسد. تاکنون الگوریتم‌ها و روش‌های خوبی برای تشخیص حملات شیلینگ ارائه شده است اما برخی از این روش‌ها یا ماتریس امتیازدهی را از یک دیدگاه واحد بررسی می‌کنند و یا از تعاملات مرتبه پایین و یا تعاملات مرتبه بالا استفاده می‌کنند. با توجه به این مورد این مقاله سازوکاری را با استفاده از ماتریس امتیازدهی کاربران، زمان امتیاز دهی کاربران و خروجی تحلیل شبکه‌های اجتماعی پروفایل کاربران با استفاده از شبکه عصبی گاوسی راف به منظور استفاده همزمان از تعاملات مرتبه پایین و مرتبه بالا برای تشخیص حملات شیلینگ ارائه می‌کند. در نهایت، ما چندین آزمایش را با سه مدل حمله میانگین، حمله تصادفی و حمله باندواگن انجام می‌دهیم و با استفاده از معیارهای دقت، فراخوانی و معیار-F1 با روش‌های PCA، Semi، BAY و XGB مقایسه می‌کنیم. نتایج نشان می‌دهد که روش پیشنهادی از روش‌های مقایسه از نظر تشخیص حمله و تشخیص کلی موثرتر عمل می‌کند، که کارایی روش ما را ثابت می‌کند.