Journal of Ambient Intelligence and Humanized Computing (2023) 14:697–710
https://doi.org/10.1007/s12652-021-03320-8

ORIGINAL RESEARCH

ShillDetector: a binary grey wolf optimization technique for detection of shilling profiles

Saumya Bansal[1] · Niyati Baliyan[1]

Received: 23 October 2020 / Accepted: 24 May 2021 / Published online: 2 June 2021

Abstract

Collaborative Filtering, though a successful recommendation technique is vulnerable to shilling attacks due to its open nature. These attacks alter recommendations being generated for the user by inserting fake user profiles in the database. To minimize the bias introduced in the recommendation process, many machine learning methods have been explored and shown excellent results. However, supervised machine learning detection techniques are restricted to hand-designed features while unsupervised detection techniques require prior knowledge about fake profiles. In this paper, we propose a novel approach namely, ShillDetector for the detection of shilling attacks based on the recently proposed swarm intelligence technique, grey wolf optimization. The proposed approach works as a dimensionality reduction technique taking advantage of high correlation among shillers and removing correlated features that are redundant. Further, it works directly on the rating matrix, does not require hand-designed features, prior knowledge of attack profiles, or any training time. The performance of ShillDetector has been evaluated on the MovieLens dataset consisting of 100 K ratings. Experimental results depict that ShillDetector outperformed two state-of-the-art approaches, namely, SVM-TIA and PCA-VarSelect approaches with an average precision of 0.99 in case of average attack taken over different attack sizes, viz, 1%, 2%, 5%, and 10%.

Keywords Recommender system · Collaborative filtering · Shilling attacks · Swarm intelligence · Evolutionary approach

1 Introduction

With a large amount of data available over the web, it becomes difficult for the user to process the data and find the relevant information from it. For instance, to watch a web series on Prime, a user might have to go through a large number of trailers before reaching a web series of interest, which is a time-consuming process and may even end up not watching any series. To help the user find the relevant information in a short time, a tool namely, Recommender System (RS) has been developed by scientists/researchers (Jannach et al. 2010). Collaborative Filtering (CF) is a memory-based RS technique that filters out items based on the interest of similar users/items (Bansal and Baliyan 2019a, b; Bedi et al. 2017; Bansal and Baliyan 2020). It is the most successful recommendation technique used by big giants namely, Amazon and Netflix. 60% of videos watched on YouTube and 40% of apps installed from the Play Store are results of recommendations.[1]

CF though successful in the world of the web is vulnerable to profile injection attacks due to the reliance of recommendations on user profiles and its open nature (Lam and Riedl 2004). Profile injection attacks are also known as shilling attacks (Burke et al. 2015). The attackers while mounting these attacks take the advantage of dependency of recommendations on other user's reactions. The fake user profiles similar to the target user are created and inserted in the dataset by the attacker to make them appear in the neighborhood and thus bias the recommendation process. Such user profiles are created by following different attack models namely, segment attack, bandwagon attack, random attack, and average attack. The purpose of such attacks is to promote or demote items for fun and profit that would otherwise may not appear in the user's list of recommended items.

Several supervised and unsupervised detection techniques have been investigated by researchers to filter out

✉ Niyati Baliyan
niyatibaliyan@igdtuw.ac.in

Saumya Bansal
saumyab271@gmail.com

[1] Department of Information Technology, Indira Gandhi Delhi Technical University for Women, New Delhi 110006, India

[1] https://developers.google.com/machine-learning/recommendation.

user profiles that can generate bias in the recommendation process. However, both techniques have certain demerits. Supervised detection techniques require a large amount of labeled data and a balanced number of fake and genuine user profiles to train the classifier (Zhou et al. 2016). Further, hand-designed features are used to train machine learning classifiers which are difficult to extract (Zhou et al. 2020). While unsupervised techniques require less computational time as unlabeled training samples are used but usually require some knowledge about shilling profiles which is difficult to find in the real world (Zhou et al. 2016, 2020). To the best of our knowledge, Swarm Intelligence (SI) techniques have not been explored by the researchers to detect fake profiles mounted in the dataset. For the ease of use and excellent results shown by bio-inspired SI technique, Grey Wolf Optimizer (GWO) on various problems including parameter tuning, economy dispatch, classification, clustering, power engineering to name a few (Hassan and Zellagui 2018; Pradhan et al. 2018; Hatta et al. 2019), we explored it from the perspective of detecting attack profiles mounted in the dataset. Further, the detection of shillers can be seen as a binary classification problem on which GWO has shown significant results in the past (Emary et al. 2016).

In this paper, we develop an unsupervised detection technique, ShillDetector for finding attack profiles mounted in the dataset. It works directly on the rating matrix, does not require hand-designed features or prior knowledge of attack profiles. Further, it shows significant detection accuracy when tested on the MovieLens (ML)[2] dataset. ShillDetector is a GWO based technique that takes inspiration from the social hierarchy of grey wolves and works on the lines of their hunting behavior, i.e., to encircle the prey before attacking it. The ease of implementation, the involvement of minimal parameters, simplicity of the algorithm, use of few operators, derivation-free nature, and excellent results (Mirjalili et al. 2014), make it more noticeable to be explored in the future by researchers. To the best of our knowledge, no meta-heuristic technique till now has been proposed for the detection of attack profiles in recommender systems.

The major contributions of the work are:

1. A novel GWO based technique for the detection of shilling attacks (ShillDetector) is proposed.
2. It works directly on the rating matrix, does not require hand-designed features or prior knowledge of attack profiles.
3. It mimics the hunting behavior of grey wolves to detect fake profiles.
4. The technique uses group behavior of attack profiles.
5. ShillDetector detects fake profiles with an average precision of 99%.
6. The simplicity of the algorithm, ease of implementation, derivation-free nature, use of fewer operators as opposed to the evolutionary algorithm (crossover, mutation), and excellent results, make it more noticeable to be explored in the future by researchers.

The paper is structured as follows: the literature review is discussed in Sect. 2. The background is discussed in Sect. 3. The proposed work is detailed in Sect. 4. Section 5 throws light on experiments and results. Section 6 concludes the work.

2 Literature review

Defending and attacking a system is a two-player game with each player's motive being 'to win'. The defender's win is in making the attack expensive, reducing the system's vulnerability, minimizing the attacker's chance of a return, and generating a robust system. On the other hand, the attacker's win is in successfully exploiting the vulnerability of the system, inserting shillers, and generating bias in the system's functionality.

To detect attack profiles mounted by the attacker in the database, various supervised and unsupervised shilling detection techniques have been discussed in the literature. A supervised detection technique using two attributes namely, Weighted Degree Agreement (WDA) and Filler Mean Target Difference (FMTD) has been proposed by Mobasher et al. (2005). Batmaz et al. (2020) proposed a technique that uses six generic and four model-specific attributes and employs kNN and SVM for classification. Cao et al. (2018), on the other hand, proposed an outlier degree detection algorithm based on dynamic feature selection. Zhou et al. (2016) proposed a two-phase SVM-TIA detection method using the Borderline-SMOTE method to balance the number of attack profiles in the training set to get rough detection results in phase-1. The target items are analyzed from attack profiles in phase-2. Supervised detection methods based on deep learning are proposed in Tong et al. (2018) and Zhou et al. (2020) considering 1 layer and 2 layer each for convolution and pooling, respectively. The basis of many unsupervised detection methods is clustering with the purpose to detect a group of attack profiles instead of a single attack profile (Mehta 2007; Mehta et al. 2007; Mehta and Nejdl 2009). Chirita et al. (2005) introduced Rating Deviation from Mean Agreement (RDMA) considering rating deviations between profiles. Few variations of PCA explored by authors are combining PCA with data complexity (Zhang et al. 2018a) and PCA with perturbation (Deng et al. 2016). Liu et al. (2019) proposed another unsupervised method using a Kalman filter based on time while Zhang et al. (2018a, b)

[2] https://grouplens.org/datasets/movielens/.

exploited user's suspicious degree based on past behavior using the hidden Markov model and hierarchical clustering.

GWO is a SI technique that has shown various applications in literature including—a prediction model using GWO with fuzzy sets to detect the diabetes disease at an early stage (Manikandan 2019), finding out the optimal feature set for the diagnosis of Parkinson's disease (Sharma et al. 2019), optimal feature selection (Emary et al. 2016), training multi-layer perceptron (Mirjalili 2015), dimensionality reduction keeping accuracy high (Elhariri et al. 2016) taking advantage of multi-objective characteristics of GWO (Emary et al. 2015).

Table 1 provides a summary of various detection techniques and the application of GWO in the literature.

This section discussed and highlighted several limitations of current detection techniques, such as the high cost involved in training labeled data, hand-designed features, and having certain prior knowledge of attack profiles in case of unsupervised methods. Further, GWO being multi-objective, i.e., it reduces dimensionality and maximizes classification accuracy at the same time, has shown remarkable results on a binary classification problem (Emary et al. 2015, 2016). Detection of shilling profiles in the dataset being a binary classification problem (Wang et al. 2015) motivated us to mathematically model the social behavior of grey wolves to distinguish between genuine and fake profiles that can manipulate recommendations generated.

3 Background

3.1 Shilling attacks

The recommendations generated by the CF depend on similar users in the neighborhood of the target user. The neighborhood can be manipulated by adding fake user profiles in the database and thus generating bias in recommendations (Gunes et al. 2014). This is termed as shilling attack or profile injection attack and is mounted with the intent of promoting or demoting an item.

From the attacker's perspective, the best attack is one that requires a minimum amount of information about the dataset, demands minimum effort, and maximizes the similarity between the shilling and genuine profiles. Taking into consideration these aspects, we have chosen average, bandwagon, random, and segment attack models among the six well-known shilling attack models (Batmaz et al. 2020), for mounting fake profiles in the database. The attack profiles are generated following the attack models described below:

1. Random attack

   Random attack is a low-knowledge attack. In order to mount such an attack, the mean of all ratings in the dataset is required (Mobasher et al. 2005; Bilge et al. 2014).

2. Average attack

   The average attack is a high-knowledge attack that proves to be successful even with a smaller filler size and can be used as a push or nuke attack (Burke et al. 2015). The average rating of each item is required by the attacker to mount such an attack.

3. Bandwagon attack

   It is a low-knowledge attack that is almost as successful as an average attack but does not need information about the mean of each item and thus is more practical to mount. It is based on highly visible items or items that a significant number of users have rated. These items are termed as selected items (I_S) and are assigned maximum rating along with the target item (Mobasher et al. 2005; Mobasher et al. 2007; Burke et al. 2015).

4. Segment attack

   It is another low-knowledge attack that mounts the attack profiles by targeting a set of users that may be interested in the target item instead of the entire user's set thus making it more meaningful and resource-saving (Mobasher et al. 2005; Burke et al. 2015; Bansal and Baliyan 2019a, b). The attack model resembles that of a bandwagon attack. For experimentation purposes, we have considered the horror movie segment. All users who have given a rating of 3 or higher, to at least 4 horror movies form a group of segment users.

The analysis in Bilge et al. (2014) depicts an increase in prediction shift with increasing filler size in case of Discrete Wavelet Transform (DWT)—based Privacy Preserving Collaborative Filtering (PPCF) for an average attack due to the transformation of successive items together. On the other hand, in case of k-means clustering-based PPCF, as filler size grows average attack becomes less successful (Bilge et al. 2014). In general, users rarely provide ratings to items, leaving most items unrated in a genuine user profiles, resulting in high sparsity in the dataset. Keeping the filler size high increases number of ratings in attack profiles and thus increases the chances of attack profiles to be less similar to authentic users (Sundar et al. 2020). Furthermore, in the case of average attack, efforts required to retrieve the mean rating of each item increases with increase in filler size (Mobasher et al. 2007). In addition, even with small filler size, average attack can prove to be just as successful (Mobasher et al. 2005). Therefore, the filler size, i.e., 1%, 3%, 5%, and 7% for all four attacks is chosen taking into consideration the knowledge efforts and sparsity of the dataset, keeping most items unrated in the attack profiles similar to genuine profiles.

The attack model varies slightly depending upon the type of attack (Gunes et al. 2014) as described in Table 2.

Table 1 Summary of detection techniques and application of GWO

Technique	Work	Description	Metrics	Dataset	Remarks
Supervised detection	Mobasher et al. (2005)	WDA and FMTD were used for detecting segment attack C4.5 is used to build a binary profile classifier	Prediction shift Hit ratio	ML 100 K	It is based on hand-designed features that are difficult to extract
	Zhou et al. (2016)	A 2-phase SVM-TIA detection method using the Borderline-SMOTE Attack profiles are detected in phase-1 Target items are detected in phase-2	False positive rate Precision Recall	ML 100 K	Lacks effective results in terms of recall under small attack size
	Tong et al. (2018)	A deep learning-based detection method is proposed 1 convolutional and 1 pooling layer is used	F-measure	ML 100 K Netflix	Shows poor performance on large scale datasets
	Cao et al. (2018)	An entropy-based algorithm to dynamically select metrics to calculate the user's outlier degree	Accuracy	ML 100 K	More feature indicators can be exploited
	Zhou et al. (2020)	2 convolutional and 2 pooling layers in the deep learning model is used for the detection of fake profiles	Precision Recall F-Measure	ML 10 M ML 20 M	Learns directly from the rating matrix but incurs a huge cost for training a large number of samples for large datasets
	Batmaz et al. (2020)	k-NN and SVM using 6 generic and 4 model-specific attributes is used	Precision Recall	ML 100 K	Focuses on detection of shillers in binary data only
Unsupervised detection	Chirita et al. (2005)	RDMA is introduced considering rating deviation between profiles	Prediction Difference	ML 100 K	RDMA can be improved further
	Mehta (2007), Mehta et al. (2007), Mehta and Nejdl (2009)	Detect a group of attack profiles instead of a single attack profile using PCA and PLSA	Precision Recall	ML 100 K	Requires prior knowledge of the number of shilling profiles
	Deng et al. (2016)	Combines PCA with perturbation to correctly classify some profiles on boundary	F-Measure	ML 100 K	Correctly identify profiles on the boundary

Table 1 (continued)

Technique	Work	Description	Metrics	Dataset	Remarks
Unsupervised detection	Zhang et al. (2018a)	Combines PCA with data complexity to overcome the drawback of Mehta (2007) and Mehta and Nejdl (2009)	F-Measure	ML 100 K	Cut-off k cannot be much greater than the attack size to achieve effective results Error on boundary profiles
	Zhang et al. (2018b)	Exploits user's suspicious degree using hidden Markov model and hierarchical clustering	Precision Recall F-measure	ML 1 M Netflix Amazon	Fails when the test set consists of purely genuine or fake profiles
	Liu et al. (2019)	Kalman filter based on time is used to detect attack profiles	Precision Recall	ML 100 K	Works well with average, random, and bandwagon attack More attacks can be explored
Grey wolf optimizer application	Emary et al. (2015)	GWO has multi-objective characteristics	Fitness function	Breast Cancer Exactly Zoo	Performs better than PSO and GA
	Mirjalili (2015)	GWO is used in training multi-layer perceptron	Mean squared error	Iris Balloon Breast Cancer	Effective in training data Shows high accuracy
	Elhariri et al. (2016)	Proposes a GWO based SVM for dimensionality reduction keeping accuracy high	Accuracy ROC	EMG Physical Action Dataset	Outperforms SVM for dimensionality reduction
	Emary et al. (2016)	Proposes a binary version of GWO for optimal feature selection	Accuracy Standard deviation F-Score	Breast Cancer Exactly Zoo	Outperform GA and PSO
	Manikandan (2019)	A diabetes prediction model to predict the disease at an early stage using GWO with fuzzy sets is proposed	Accuracy Precision Recall	PIMA dataset related to diabetes prediction	Outperforms ant colony optimization with fuzzy sets
	Sharma et al. (2019)	An optimal feature set for the diagnosis of Parkinson's disease is obtained	Accuracy Detection rate False alarm rate	Hand PD Speech PD Voice PD	Shows high accuracy in prediction of Parkinson disease

Table 2 Attack models

Attack model	I_S	I_F	I_ϕ	I_T
Random	NA	Rating around overall mean of rating matrix	0	Maximum rating
Average	NA	Mean rating of item across users	0	Maximum rating
Bandwagon	Maximum rating	Rating around overall mean of the rating matrix	0	Maximum rating
Segment	Maximum rating	Minimum rating	0	Maximum rating

I_S refers to set of selected items with particular characteristics to make shillers similar to a genuine profile, I_F refers to set of filler items chosen randomly to complete attack profile, I_ϕ is set of unrated items, I_T is the target item

3.2 Basic grey wolf optimization model

GWO proposed by Mirjalili et al. (2014) is a SI technique that mimics the social behavior of grey wolves to capture the prey. Grey wolves live in a group of 5–12 and have a strong dominance hierarchy. Some of the advantages of GWO (Hatta et al. 2019; Emary et al. 2015; Mirjalili et al. 2014; Faris et al. 2018) are: simplicity, ease to operate, few operators unlike the genetic algorithm (crossover, mutation, and so on), and a high convergence rate. Further, it is flexible i.e. can be applied in various applications such as optimization, power engineering, bioinformatics, image processing, etc. To leverage the above-mentioned benefits of GWO, a huge volume of work has been done on applying GWO in solving problems of various domains. In the mathematical model of GWO, the fittest solution is α followed by β and so on as in the hierarchy shown in Fig. 1 (Mirjalili et al. 2014).

GWO starts by assigning random positions to grey wolves (search agents). The fitness function is used to compute the fitness value of each search agent based on the current position. Throughout iteration, α, β, and δ are assigned best positions (closest to prey) and other search agent's positions are updated accordingly. The components of $\vec{a}$ are linearly decreased from 2 to 0 throughout iterations (Mirjalili et al. 2014). This process is repeated till the termination condition is reached. The pseudocode for GWO is given below:

Pseudocode of GWO

```
Notations:
  • $\vec{a}$ : Co-efficient vector decreasing linearly from 2 to 0
  • $max\_iter$ : maximum number of iterations
Randomly initialize positions of grey wolves
Calculate the fitness of each grey wolf based on the current position
Assign best, second best, and third best solution to vectors $\overrightarrow{X_\alpha}, \overrightarrow{X_\beta}, \overrightarrow{X_\delta}$ respectively based on fitness value
while $l$ in range (max_iter)
    Update $\vec{a} = 2 - l \times \frac{2}{\text{max\_iter}}$
    where $l \in [0, \text{max\_iter}]$
    for each grey wolf
        Update position of grey wolf based on $\overrightarrow{X_\alpha}, \overrightarrow{X_\beta}, \overrightarrow{X_\delta}$
        Calculate fitness of all wolves
    Update $\overrightarrow{X_\alpha}, \overrightarrow{X_\beta}, \overrightarrow{X_\delta}$
return $\overrightarrow{X_\alpha}$
```

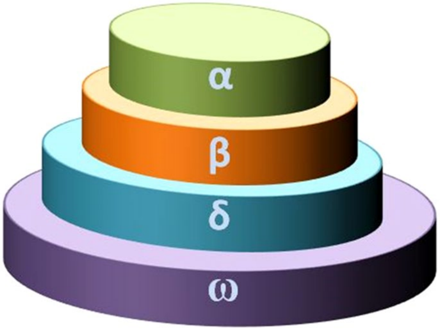
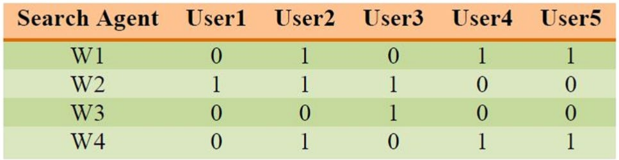


Fig. 1 Social hierarchy of grey wolves



Search Agent	User1	User2	User3	User4	User5
W1	0	1	0	1	1
W2	1	1	1	0	0
W3	0	0	1	0	0
W4	0	1	0	1	1

Fig. 2 Random initialization of 4 search agents (wolves) and 5 users

4 Proposed work

4.1 Motivation

The trust and reliability of the user on recommendations generated are extremely substantial for the continuity of the system. Malicious users may compromise with the trust and reliability of recommendations by injecting fake user profiles in the database. Therefore, the purpose is to nullify/minimize the effect of fake profiles on recommendations generated. There exists a high correlation among shillers due to the same underlying model used to generate them (Mehta et al. 2007). Therefore, the detection of shillers can be seen as a dimensionality reduction problem and thus minimizing the redundancy that exists in the database. GWO has the capability of solving bi-objective problems i.e. dimensionality reduction keeping high classification accuracy. Further, it has been used for feature selection in various applications of machine learning (Al-Tashi et al. 2020). But, to the best of our knowledge, no work till now has used GWO for the detection of shillers in RS. Further, detection of fake profiles can be seen as a binary classification problem: 1 for a genuine profile and 0 for a fake profile. Considering this, a binary version of GWO has been used in the detection of shilling attacks.

1. Proposed approach

   This subsection details the proposed algorithm (ShillDetector) for the detection of fake profiles in the database following the attack models namely, average attack, bandwagon attack, and segment attack. The ShillDetector takes advantage of the application of GWO i.e. feature reduction (Emary et al. 2015). Further, the algorithm is explained in a step-wise fashion:

Table 3 User-item rating matrix

	I1	I2	…	IM
U1	5	3	…	?
U2	?	4	…	4
…	2	5	…	3
UK	4	?	…	2

2. Pre-processing phase

   The dataset is transformed into a user-item rating matrix (R) consisting of M items and K users as shown in Table 3. Here, ? denotes an item not seen/not rated by the user.

3. Clustering of users

   In this step, clusters of users are created based on Pearson correlation among users using k-Means. Next, we find the top-N highly correlated users based on the Pearson correlation coefficient computed. Finally, the cluster number containing the maximum number of top-N highly correlated users is returned which is used in a later stage of the proposed approach. This step is based on the hypothesis that fake profiles have a higher correlation among them as compared to genuine profiles (Mehta and Nejdl 2009). Therefore, the cluster number returned will be containing most of the shillers. However, the resultant cluster may contain genuine profiles also.

4. Transpose of matrix

   ShillDetector is a dimensionality reduction technique that considers users as features. Therefore, we transpose the user-item rating matrix to store users as columns instead of rows i.e. R^T.

5. Feature importance computation

   We compute the importance of each feature (user) by importing *feature_importance* attribute of the random forest regressor from *sklearn.ensemble*. The intuition behind this step is to get a low feature importance value for highly correlated users as such users do not contribute much to the functionality of any system and are thus considered redundant. The importance of each feature in *feature_importance* attribute is computed based on the feature's contribution in determining the split. The aggregation of the importance of all features is 1 with each user's importance between 0 and 1.

6. Mathematical computation on lines of GWO

   This step is built following the original GWO model (Mirjalili et al. 2014).

a. We first initialize vectors and variables required in ShillDetector.

i. α_{pos},β_{pos} and $\delta_{pos} = \vec{0}$. They are binary vectors of size (1 × *n_users*) where *n_users* represents the total number of users in the dataset. 1 in binary vector represents genuine profile whereas 0 represents a fake profile. Among all search agents, three search agents nearest to prey are termed as α, β, δ, and their current position is stored in α_{pos} ,β_{pos} and δ_{pos} respectively.
α_{score},β_{score} and δ_{score} represents the fitness score of α, β, and δ. They are initialized to 0.
ii. Randomly initialize the position of all search agents (grey wolves) who live in a pack of 5–12. The position of each search agent is represented using one-dimensional binary vector of size (1 × *n_users*). Figure 2 shows an instance of a randomly initialized position of search agents where 1 signifies the genuine profile and 0 signifies the fake profile.

b. Next, the fitness of each search agent is computed using the objective function as described by Eq. (1).

$$\text{Maximize fit (i)} = (\alpha \times \text{agg\_imp\_feature [i]}) + \left(\beta \times \frac{\text{selected\_features [i]}}{\text{total\_features}}\right)$$
$$\text{Subject to}$$
$$\text{Constraints } \alpha, \beta = 0.5 \text{ to mark the balance between two,} \tag{1}$$

where *i* represents the search agent; *selected_features*[*i*] refers to the total number of 1's in the vector of search agent; agg_imp_feature[*i*] represents aggregation of the importance of features of search agent computed in step "4 Feature importance computation"; total_features is the total number of features in the dataset.

In agg_imp_feature[i], the importance of all features enabled in search agent, i.e., 1 (genuine profile) and not belong to selected cluster in step 2 is added along with the importance of all features disabled in search agent, i.e., 0 (fake profile) and belong to the selected cluster. Here, the selected cluster contains highly correlated users based on the hypothesis that fake profiles having a higher correlation among them as compared to genuine profiles. To sum up, we aggregate the importance of all features that are being correctly identified by the search agent.

c. Find the fittest (best) search agent, i.e., search agent with maximum fitness value and assign the position vector and score to α_{pos} and α_{score} respectively. Similarly, find second and third fittest search agents and assign values toβ_{pos}, β_{score} andδ_{pos}, δ_{score}, respectively.

d. Update $\vec{a}$ which is used in the sub-point 'e'.

$$\vec{a} = 2 - 1 \times \frac{2}{\text{max\_iter}} \tag{2}$$

where $\vec{a}$ is a co-efficient vector; max_iter refers to maximum number of iterations; 1 ranges from 0 to max_iter

e. The position of each search agent is updated using Eq. (3)—Eq. (5) taking inspiration from the original GWO encircling process.

$$\begin{aligned} D_\alpha &= \left|\left(C1 \times \alpha_{pos}[j]\right) - \text{position}[i][j]\right|; \\ X1 &= \alpha_{pos}[j] - (A1 \times D_\alpha) \end{aligned} \tag{3}$$

$$\begin{aligned} D_\beta &= \left|\left(C2 \times \beta_{pos}[j]\right) - \text{position}[i][j]\right|; \\ X2 &= \beta_{pos}[j] - (A2 \times D_\beta) \end{aligned} \tag{4}$$

$$\begin{aligned} D_\delta &= \left|\left(C3 \times \delta_{pos}[j]\right) - \text{position}[i][j]\right|; \\ X3 &= \delta_{pos}[j] - (A3 \times D_\delta) \end{aligned} \tag{5}$$

$$X = \frac{X1 + X2 + X3}{3} \tag{6}$$

where A1, A2, and A3 are coefficient vectors computed using Eq. (7); C1, C2, and C3 are coefficient vectors computed using Eq. (8); D_α, D_β, D_δ, $X1, X2 \text{and} X2$ are vectors; position[i][j] is the value of search agent 'i' for feature 'j'; $\alpha_{pos}[j]$, $\beta_{pos}[j]$, $\delta_{pos}[j]$ represent positional value for feature 'j'.

$$A1, A2, A3 = 2\vec{a}.\vec{r_1} - \vec{a} \tag{7}$$

$$C1,\ C2,\ C3 = 2.\vec{r}_2 \tag{8}$$

where $\vec{r_1}$ and $\vec{r_2}$ are random vectors in the range [0,1].

Update the position vector of the search agent by finding a sigmoid of X taking inspiration from the hunting step of GWO.

f. Repeat step 5 (sub-point 'b' to 'e') till max_iter is reached or algorithm converges i.e. there is no improvement over the past two iterations.

7. Use α_{pos} for the detection of fake profiles from the dataset.

Table 4 Parameters and their value

Parameter	Value	Remarks
Max_iter	100	Number of iteration to get the best solution
No of clusters	10	Gives the best solution in each case considered
Search Agents	12	Grey wolf lives in a pack of 5–12
A	0.5	To mark balance between both parts of Eq. (1)

5 Experiments and results

In this section, the dataset is discussed followed by experiments and results.

5.1 Dataset and experimental methodology

For experimentation purposes, a publicly available ML[3] dataset of size 100 K has been used. The user rates the movie on a scale of 1–5 giving a rating to at least 20 movies. The users corresponding to ML 100 K dataset are considered genuine while fake profiles/shillers are added to the dataset using the attack model. Different attack and filler sizes have been considered for generating attack profiles keeping the target item constant for experimental purposes. The proposed approach is an unsupervised technique and therefore does not require any additional training time.

5.2 Parameter setting

Several parameters need to be defined while implementing and analyzing ShillDetector.

5.2.1 Fixed parameter

There are a few parameters that need to be initialized and remain the same for every experiment of ShillDetector as mentioned in Table 4. Further, seeking the advantages of GWO, only 2 hyperparameters (a and c) that helps the learning process have to adjust.

5.2.2 Varied parameters

The attack size and filler size are two parameters that are being varied to investigate the proposed approach.

a. Attack size

It is defined as a ratio of fake profiles to the total number of profiles. The attack size ranging from 1 to 30% is being considered for experimentation purposes taking into consideration information and efforts of the attacker to mount the attack.

b. Filler size

It is defined as the ratio of ratings provided in the user profile to the total number of items in the dataset (Zhou et al. 2016). Taking into consideration the sparsity of the dataset, filler size is usually kept small i.e. 1%, 3%, 5%, and 7% make the fake profile resemble genuine profiles.

5.3 Evaluation metrics

To evaluate the performance of the proposed approach, several standard metrics have been used (Sharma et al. 2019; Al-Tashi et al. 2019). Each evaluation metrics computes the average of M runs where M is taken to be 10.

a. Classification accuracy

It indicates the correctness of ShillDetector in classifying fake and genuine profiles.

$$\text{Classification accuracy} = \frac{\mathrm{X}}{\text{Total\_profiles}} \times 100 \tag{9}$$

where X indicates the number of correctly classified profiles.

b. Detection rate

It signifies the % of correctly identified fake profiles.

$$\text{Detection rate} = \frac{\mathrm{Y}}{\text{total\_fake\_profiles}} \times 100 \tag{10}$$

where Y is the number of fake profiles correctly identified.

c. False Alarm Rate (FAR)

It counts the number of genuine profiles classified as fake.

$$\mathrm{FAR} = \frac{\mathrm{FP}}{\text{genuine\_profiles}} \times 100 \tag{11}$$

where FP is the number of genuine profiles misclassified as fake.

d. Precision

It is defined as the number of fake profiles correctly classified to the number of profiles classified as fake.

e. Recall

It is the number of fake profiles correctly identified to the number of profiles.

5.4 Experiments and results

This subsection reports and discusses the results obtained by conducting several experiments from various perspectives.

[3] https://grouplens.org/datasets/movielens/

Table 5 Investigation results of ShillDetector using different attack models on ML 100 K

Evaluation metric		Filler size (%)	Attack size						
			1%	2%	5%	10%	15%	20%	30%
Classification accuracy	Average attack	1	99.97	100	100	99.86	99.93	99.99	98.94
		3	99.94	100	100	99.98	99.95	100	99.71
		5	99.71	100	100	99.97	100	99.97	99.77
		7	99.79	99.94	99.84	99.92	99.88	99.91	99.91
	Bandwagon attack	1	99.86	100	100	99.95	99.78	99.97	99.73
		3	99.89	100	100	99.97	99.95	99.98	99.56
		5	99.94	100	100	100	100	99.91	99.74
		7	99.63	99.87	99.82	99.87	99.88	99.91	99.91
	Segment attack	1	99.92	99.98	100	100	99.80	99.78	99.75
		3	99.92	100	100	99.98	100	99.92	99.57
		5	99.92	100	99.92	99.93	99.94	99.93	99.47
		7	99.89	99.55	99.92	99.77	99.89	99.92	99.94
Detection rate	Average attack	1	97.50	97.50	98.93	98.27	99.55	99.53	97.60
		3	97.75	98.12	99.20	99.33	98.93	99.80	99.02
		5	97.75	97.50	99.46	99.33	99.73	99.40	99.02
		7	100	100	98.40	99.73	99.29	99.47	99.64
	Bandwagon attack	1	97.50	96.87	98.67	99.20	98.75	99.53	98.93
		3	92.50	98.75	98.67	99.20	99.20	99.53	98.62
		5	95.00	96.25	98.40	99.33	99.64	99.13	98.89
		7	95.00	96.05	98.40	99.20	99.11	99.60	99.64
	Segment attack	1	97.50	96.25	99.46	99.20	98.58	98.80	98.93
		3	97.50	96.87	99.20	99.06	99.55	99.46	98.98
		5	97.50	96.25	98.13	99.06	99.64	99.46	98.13
		7	95.00	94.73	98.93	99.12	99.32	99.80	99.46
False alarm rate	Average attack	1	0.00	0.03	0.01	0.07	0.11	0.01	0.25
		3	0.02	0.01	0.00	0.05	0.00	0.03	0.18
		5	0.21	0.00	0.02	0.06	0.00	0.01	0.10
		7	0.02	0.05	0.07	0.05	0.02	0.00	0.00
	Bandwagon attack	1	0.00	0.00	0.03	0.07	0.17	0.03	0.13
		3	0.02	0.01	0.03	0.05	0.03	0.02	0.25
		5	0.00	0.01	0.01	0.02	0.03	0.03	0.10
		7	0.03	0.05	0.10	0.05	0.00	0.02	0.00
	Segment attack	1	0.00	0.03	0.06	0.02	0.11	0.11	0.10
		3	0.05	0.00	0.00	0.02	0.02	0.09	0.35
		5	0.05	0.01	0.09	0.07	0.11	0.07	0.22
		7	0.05	0.03	0.02	0.0.5	0.00	0.02	0.00
Precision	Average attack	1	1.00	0.98	0.99	0.99	0.99	0.99	0.97
		3	0.97	0.99	1.00	0.99	1.00	0.99	0.99
		5	0.92	1.00	0.99	0.99	1.00	0.99	0.99
		7	0.93	0.97	0.98	0.99	0.99	1.00	1.00
	Bandwagon attack	1	1.00	1.00	0.99	0.99	0.98	0.99	0.99
		3	1.00	0.99	0.99	0.99	0.99	0.99	0.99
		5	0.97	0.99	0.99	0.99	0.99	0.99	0.99
		7	0.98	0.97	0.97	0.99	1.00	0.99	1.00
	Segment attack	1	1.00	0.98	0.98	0.99	0.99	0.99	0.99
		3	0.95	0.99	0.98	0.99	0.99	0.99	0.99
		5	0.95	1.00	1.00	0.99	0.99	0.99	0.98
		7	0.95	0.98	0.99	0.99	0.99	0.99	0.99

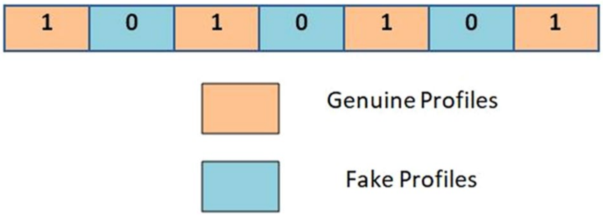


Fig. 3 Output of ShillDetector labeling users as fake/genuine

5.4.1 Binary operator used

Each user in the dataset can be classified as fake/genuine as depicted in Fig. 3. Therefore, shilling profile detection can be considered as a binary classification problem that provides a label to each user i.e. fake or genuine. The proposed approach ShillDetector uses a binary version of GWO to detect fake profiles in the dataset. We have used binary operator sigmoid(Al-Tashi et al. 2019) to transform GWO into a binary version to fit the problem of feature selection.

5.4.2 Result analysis

The performance of ShillDetector has been analyzed by conducting various experiments and results have been tabulated in Table 5. Filler size plays a crucial role in creating attack profiles and thus generating bias in recommendations. To fill up ratings of filler items in attack profiles created using average attack, a huge amount of information about the rating distribution (mean rating for every item) is needed by the attacker which is often difficult to extract and incurs huge efforts. Therefore, the filler size is kept small. Another reason to keep the filler size small is the sparsity of the dataset. As most items remain unrated in a genuine user's profile,

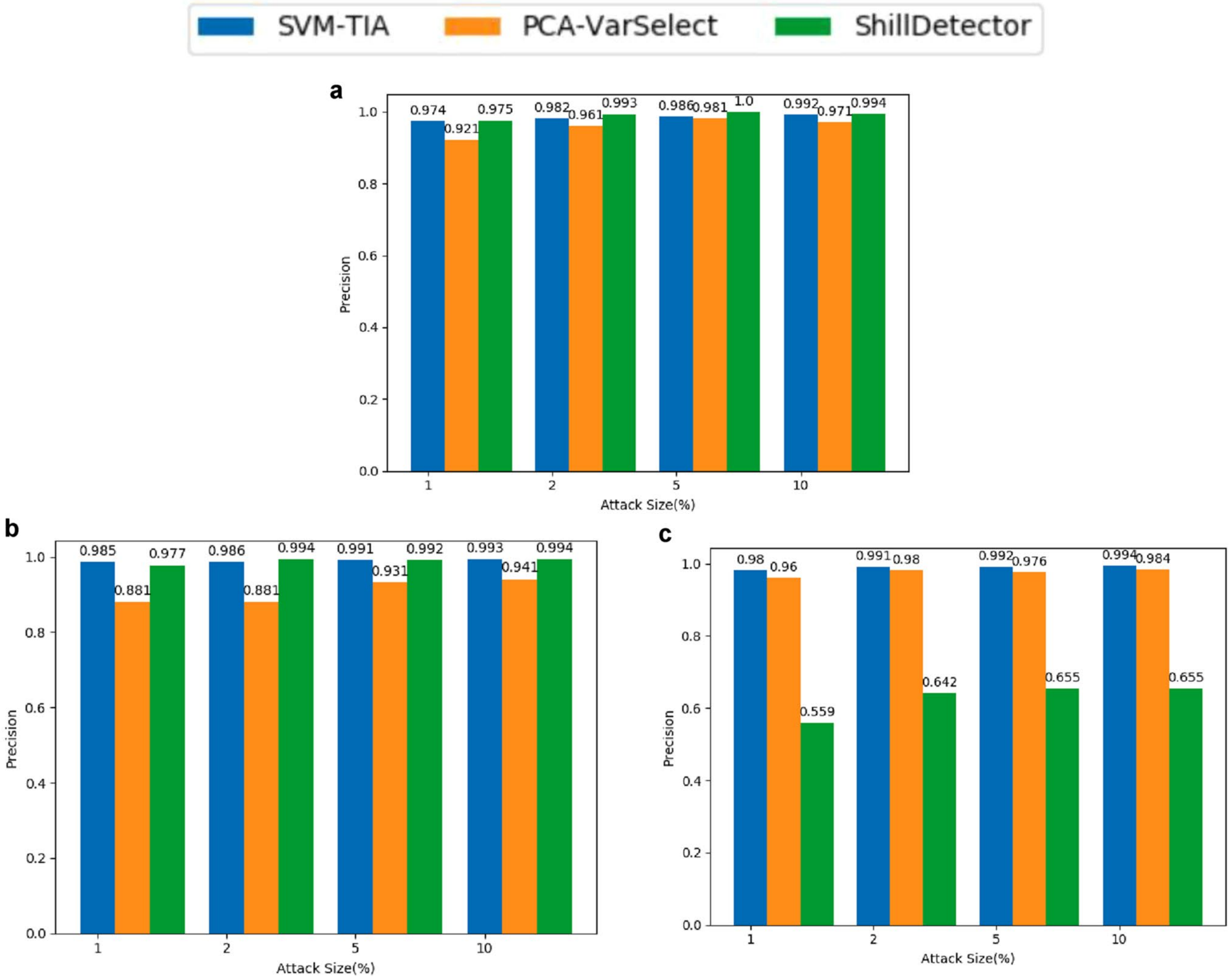


Fig. 4 Comparison of ShillDetector with state-of-the-art approaches in terms of precision. **a** Comparative analysis of average attack. **b** Comparative analysis of bandwagon attack. **c** Comparative analysis of random attack

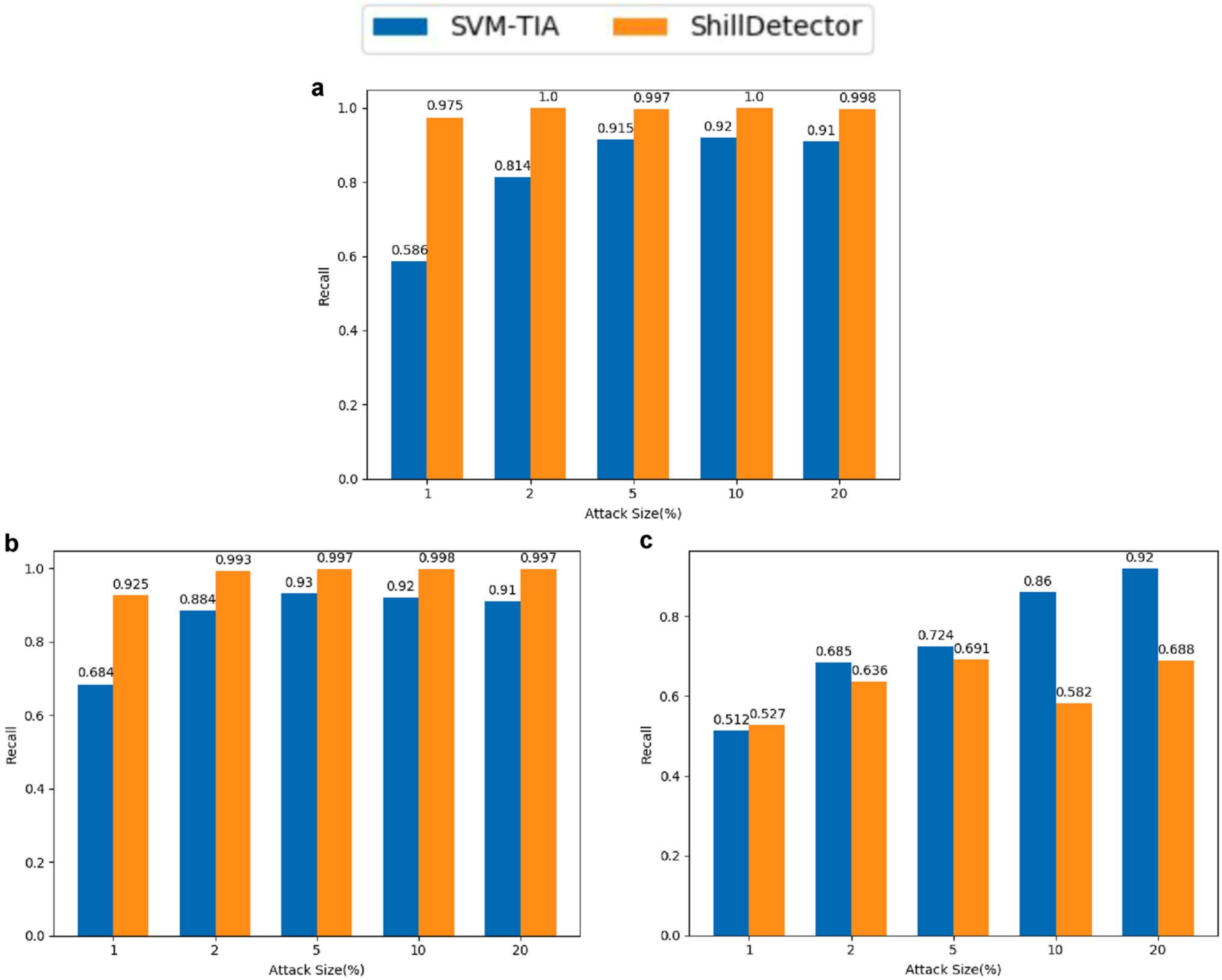


Fig. 5 Comparison of ShillDetector with SVM-TIA in terms of recall. **a** Comparative analysis of average attack. **b** Comparative analysis of bandwagon attack. **c** Comparative analysis of random attack

filler size is kept small i.e. 1%, 3%, 5%, and 7% to keep most items unrated in the attack profile too.

Investigations have been done by mounting different attacks considering different attack sizes. It is worth noting that, the classification accuracy of ShillDetector is above 99% in almost all cases depicting the correct classification of fake and genuine profiles. Further, a high detection rate i.e. above 95% is shown in all cases considered except at filler size 3% and attack size 1% for Bandwagon attack. However, the detection rate of 92.5% in such a case signifies the misclassification of 1 out of 10 fake profiles. The small FAR i.e. below 0.25 in most cases has been observed. However, in the case of filler size 3% and attack size 30%, FAR as high as 0.35 has been observed signifying misclassification of one genuine profile on an average. To sum up, it can be inferred that at max only one user profile (genuine/fake) has been misclassified by ShillDetector depicting its excellent performance.

ShillDetector shows excellent detection results in case of strong attacks such as average, bandwagon, and segment attack on different filler sizes ranging from 1 to 7% seeking to the high correlation between profiles due to underlying attack models.

5.4.3 Comparative analysis

In this subsection, a comparative analysis of ShillDetector with two other state-of-the-art approaches, namely, SVM-TIA (Zhou et al. 2016) and PCA-VarSelect (Mehta and Nejdl 2009) is drawn on the ML-100 K dataset. SVM-TIA is a variant of SVM that has been primarily explored for classification. On the other hand, PCA-VarSelect has originated from

PCA which is a clustering technique. Both techniques have shown good accuracy in the detection of shilling attacks but have certain drawbacks. SVM-TIA requires target item analysis which is sometimes difficult to find when a large number of items are available. The performance of SVM-TIA becomes unstable in such situations. Further, it lacks effective results in terms of recall under small attack size. On the other hand, PCA-VarSelect requires prior knowledge of the total number of attack profiles which is infeasible in the real world. A comparison among these approaches is laid using precision and recall. Bandwagon, average and random attack model were considered and filler size of 3% taking a similar underlying part of all three approaches.

A comparison of all three approaches when attack profiles are inserted using bandwagon, average and random attack in terms of precision is shown in Fig. 4. ShillDetector outperformed PCA-VarSelect and SVM-TIA on the average attack. For bandwagon attack, at attack size, 1%, the precision of ShillDetector is found to be 0.977 which is slightly small in comparison to a precision of 0.985 for SVM-TIA. However, ShillDetector's performance is excellent in this case also as the accuracy of classification is still 99.89% as can be seen from Table 5. Only one fake profile has been misclassified. On a weak attack, such as a random attack, fake profiles do not have a high correlation among them due to the underlying model. Further, the fake profiles tend to be distributed across different clusters, thus making it difficult for ShillDetector to detect them. However, such attacks have a diminutive impact on the performance of recommendations as they are weak attacks and seldom occur in the neighborhood of the target user (Mobasher et al. 2007).

Next, a comparison between SVM-TIA and ShillDetector is drawn based on recall. The results are depicted in Fig. 5 when attack profiles are inserted using average, random, and bandwagon attack model. ShillDetector outperformed SVM-TIA on both average and bandwagon attack on all attack sizes considered with the highest recall value of 1 and the lowest recall value of 0.925. However, in the case of random attack, a lower recall value has been observed.

To sum up, ShillDetector which uses a variant of recently developed SI technique namely, GWO, is an unsupervised technique i.e. no training process required and thus saves CPU Time. Further, hand-designed features are not required, unlike the supervised technique. It is easy to operate, implement, requires few parameters, and provides results with excellent classification accuracy.

6 Conclusion

In this paper, we proposed a novel approach namely, ShillDetector for the detection of fake profiles that can be inserted by the attacker in the dataset with a motive of generating bias in the recommendation process. ShillDetector is based on the Grey Wolf Optimization technique which is a swarm intelligence technique and mimics the social behavior of grey wolves for reaching the prey. The proposed approach exploits group characteristics that exist among shillers by working directly on a user-item rating matrix. Further, it works as a feature selection technique that is easy to operate, requires no training time, and has few parameters to adjust. Further, ShillDetector can be used as a pre-processed phase of any recommendation algorithm and thus can save the recommendation process from generating biased recommendations.

Data availability statement The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

References

Al-Tashi Q, Kadir SJA, Rais HM, Mirjalili S, Alhussian H (2019) Binary optimization using hybrid grey wolf optimization for feature selection. IEEE Access 7:39496–39508

Al-Tashi Q, Rais HM, Abdulkadir SJ, Mirjalili S, Alhussian H (2020) A review of grey wolf optimizer-based feature selection methods for classification. In evolutionary machine learning techniques. Springer, Singapore, pp 273–286

Bansal S, Baliyan N (2019a) A study of recent recommender system techniques. Int J Knowl Syst Sci (IJKSS) 10(2):13–41

Bansal S, Baliyan N (2019b) Evaluation of collaborative filtering based recommender systems against segment-based shilling attacks. In: 2019 International Conference on Computing, Power and Communication Technologies (GUCON) (pp 110–114). IEEE

Bansal S, Baliyan N (2020) Bi-MARS: a bi-clustering based memetic algorithm for recommender systems. Appl Soft Comput 97:106785

Batmaz Z, Yilmazel B, Kaleli C (2020) Shilling attack detection in binary data: a classification approach. J Ambient Intell Humaniz Comput 11(6):2601–2611

Bedi P, Gautam A, Bansal S, Bhatia D (2017) Weighted bipartite graph model for recommender system using entropy based similarity measure. In: The International Symposium on Intelligent Systems Technologies and Applications (pp 163–173). Springer, Cham

Bilge A, Gunes I, Polat H (2014) Robustness analysis of privacy-preserving model-based recommendation schemes. Expert Syst Appl 41(8):3671–3681

Burke R, O'Mahony MP, Hurley NJ (2015) Robust collaborative recommendation. In: Recommender systems handbook. Springer, Boston, pp 961–995

Cao G, Zhang H, Fan Y, Kuang L (2018) Finding shilling attack in recommender system based on dynamic feature selection. In SEKE (pp 50–55)

Chirita PA, Nejdl W, Zamfir C (2005) Preventing shilling attacks in online recommender systems. In: Proceedings of the 7th annual ACM international workshop on Web information and data management (pp 67–74)

Deng ZJ, Zhang F, Wang SP (2016) Shilling attack detection in collaborative filtering recommender system by PCA detection and perturbation. In: 2016 International Conference on Wavelet Analysis and Pattern Recognition (ICWAPR) (pp 213–218). IEEE

Elhariri E, El-Bendary N, Hassanien AE (2016) Bio-inspired optimization for feature set dimensionality reduction. In: 2016 3rd International Conference on Advances in Computational Tools for Engineering Applications (ACTEA) (pp 184–189). IEEE

Emary E, Yamany W, Hassanien AE, Snasel V (2015) Multi-objective gray-wolf optimization for attribute reduction. Procedia Comput Sci 65:623–632

Emary E, Zawbaa HM, Hassanien AE (2016) Binary grey wolf optimization approaches for feature selection. Neurocomputing 172:371–381

Faris H, Aljarah I, Al-Betar MA, Mirjalili S (2018) Grey wolf optimizer: a review of recent variants and applications. Neural Comput Appl 30(2):413–435

Gunes I, Kaleli C, Bilge A, Polat H (2014) Shilling attacks against recommender systems: a comprehensive survey. Artif Intell Rev 42(4):767–799

Hassan HA, Zellagui M (2018) Application of grey wolf optimizer algorithm for optimal power flow of two-terminal HVDC transmission system. Adv Electric Electron Eng 15(5):701–712

Hatta NM, Zain AM, Sallehuddin R, Shayfull Z, Yusoff Y (2019) Recent studies on optimisation method of Grey Wolf Optimiser (GWO): a review (2014–2017). Artif Intell Rev 52(4):2651–2683

Jannach D, Zanker M, Felfernig A, Friedrich G (2010) Recommender systems: an introduction. Cambridge University Press

Lam SK, Riedl J (2004) Shilling recommender systems for fun and profit. In: Proceedings of the 13th international conference on World Wide Web (pp 393–402)

Liu X, Xiao Y, Jiao X, Zheng W, Ling Z (2019) A novel Kalman Filter based shilling attack detection algorithm. arXiv preprint arXiv:1908.06968

Manikandan K (2019) Diagnosis of diabetes diseases using optimized fuzzy rule set by grey wolf optimization. Pattern Recogn Lett 125:432–438

Mehta B (2007) Unsupervised shilling detection for collaborative filtering. In AAAI (pp 1402–1407)

Mehta B, Nejdl W (2009) Unsupervised strategies for shilling detection and robust collaborative filtering. User Model User-Adap Inter 19(1–2):65–97

Mehta B, Hofmann T, Fankhauser P (2007) Lies and propaganda: detecting spam users in collaborative filtering. In: Proceedings of the 12th international conference on Intelligent user interfaces (pp 14–21)

Mirjalili S (2015) How effective is the Grey Wolf optimizer in training multi-layer perceptrons. Appl Intell 43(1):150–161

Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. Adv Eng Softw 69:46–61

Mobasher B, Burke R, Williams C, Bhaumik R (2005) Analysis and detection of segment-focused attacks against collaborative recommendation. In: International Workshop on Knowledge Discovery on the Web. Springer, Berlin, pp 96–118

Mobasher B, Burke R, Bhaumik R, Williams C (2007) Toward trustworthy recommender systems: an analysis of attack models and algorithm robustness. ACM Trans Internet Technol (TOIT) 7(4):23

Pradhan M, Roy PK, Pal T (2018) Oppositional based grey wolf optimization algorithm for economic dispatch problem of power system. Ain Shams Eng J 9(4):2015–2025

Sharma P, Sundaram S, Sharma M, Sharma A, Gupta D (2019) Diagnosis of Parkinson's disease using modified grey wolf optimization. Cogn Syst Res 54:100–115

Sundar AP, Li F, Zou X, Gao T, Russomanno ED (2020) Understanding shilling attacks and their detection traits: a comprehensive survey. IEEE Access 8:171703–171715

Tong C, Yin X, Li J, Zhu T, Lv R, Sun L, Rodrigues JJ (2018) A shilling attack detector based on convolutional neural network for collaborative recommender system in social aware network. Comput J 61(7):949–958

Wang Y, Zhang L, Tao H, Wu Z, Cao J (2015) A comparative study of shilling attack detectors for recommender systems. In: 2015 12th International Conference on Service Systems and Service Management (ICSSSM) (pp 1–6). IEEE

Zhang F, Zhang Z, Zhang P, Wang S (2018) UD-HMM: An unsupervised method for shilling attack detection based on hidden Markov model and hierarchical clustering. Knowl-Based Syst 148:146–166

Zhang F, Deng ZJ, He ZM, Lin XC, Sun LL (2018a) Detection of shilling attack in collaborative filtering recommender system by pca and data complexity. In: 2018 International Conference on Machine Learning and Cybernetics (ICMLC) (Vol. 2, pp 673–678). IEEE

Zhou W, Wen J, Xiong Q, Gao M, Zeng J (2016) SVM-TIA a shilling attack detection method based on SVM and target item analysis in recommender systems. Neurocomputing 210:197–205

Zhou Q, Wu J, Duan L (2020) Recommendation attack detection based on deep learning. J Inf Secur Appl 52:102493

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.