Soft Computing
https://doi.org/10.1007/s00500-021-05586-8

FOCUS

Detection of shilling attack in recommender system for YouTube video statistics using machine learning techniques

Shalli Rani[1] · **Manpreet Kaur**[1] · **Munish Kumar**[2] · **Vinayakumar Ravi**[3] · **Uttam Ghosh**[4] · **Jnyana Ranjan Mohanty**[5]

Abstract

Literature survey shows that the recommendation systems have been largely adapted and evaluated in various domains. Due to low performances from various cyber attacks, the adoption of recommender system is in the initial stage of defense systems. One of the most common attacks for recommender system is shilling attack. There are some existing techniques for identifying the shilling attacks built in the user ratings patterns. The performance of ratings on target items differs between the attack user profiles and actual user profiles. To differentiate the certain profiles, the affected profiles are known as attack profiles. Besides the shilling attacks, real cyber attacks are taking place in the community which are being solved by Petri Net methods. These attacks can be falsely predicted (shilling attacks) by the users which can raise security threats. For identifying various shilling attacks without a priori knowledge, Recommendation System suffers from low accuracy. Basically, recommendation attack is split into nuke and push attack that encourage and discourage the recommended target item. The strength of shilling attack is usually measured by filler size and attack size. An experiment over unsupervised machine learning algorithms with filler size 3% over 3%, 5%, 8% and 10% attack sizes is presented for Netflix dataset. Furthermore, we conducted an experiment on data of 26 K videos on the Trending YouTube Video Statistics, to predict the user preferences for a particular genre of videos using Machine Learning Algorithms. Based on the results, it observed that the Boosted Decision tree performs the best with an accuracy of 99 percent.

Keywords Recommender system · Shilling attack · Collaborative filtering · YouTube video statistics · Machine learning · Cyber attacks · Security threats · Defense systems · Soft computing

Communicated by Suresh Chandra Satapathy.

✉ Shalli Rani
shalli.rani@chitkara.edu.in; Shallir79@gmail.com

Manpreet Kaur
manpreet.me1117123@chitkara.edu.in

Munish Kumar
munishcse@gmail.com

Vinayakumar Ravi
vinayakumarr77@gmail.com; vravi@pmu.edu.sa

Uttam Ghosh
uttam.ghosh@vanderbilt.edu

Jnyana Ranjan Mohanty
jmohantyfca@kiit.ac.in

[1] Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, Punjab, India

[2] Department of Computational Sciences, Maharaja Ranjit Singh Punjab Technical University, Bathinda, Punjab, India

[3] Center for Artificial Intelligence, Prince Mohammad Bin Fahd University, Khobar, Saudi Arabia

[4] Vanderbilt University, Nashville, USA

[5] Kalinga Institute of Industrial Technology (KIIT) Deemed To Be University, Bhubaneswar, India

Published online: 28 January 2021

1 1. Introduction

With the overloaded information over the internet, Recommender system (RS) is widely used in every field, for example Recommendations of products on e-commerce websites, recommendation of web pages on intelligent web systems, etc. There are some recommendation techniques that exist, one of them most likely used in the recommender system is collaborative filtering and has turned into an essential part of many browsing websites including Netflix, Amazon and Flipkart. However, collaborative filtering system utilizes sparse user-item matrix, i.e., n*m, which includes n number of users preferences about m products (Bilge et al. 2014). Further, developers must access multiple aspects, including the quality of recommendation and the strength of the recommender system on rating set changes.

The first aspect, Quality of recommender system, depends on the properties of recommendation it produces, which are typically accessed using performance metrics that capture a narrow view of system behavior. Most commonly, developers access the accuracy of recommendation with the metrics to measure the ability of the RS to faithfully represent a set of known preferences. Accuracy is measured by reserving a portion of a ratings dataset as a set of known preferences and using the rest of the ratings to train a recommender system and make recommendations. By computing the accuracy of recommendation algorithms with known preferences such as Recall, Precision on that dataset, the stability of RS can be assessed. The strength of the RS measures the consistency of recommendations after changes are made on the dataset. The stability of recommender systems can be an expensive property to measure without extensive knowledge of the recommender algorithm, as it requires modifications of generating dataset, modified data with training recommender and measuring the distance between the new and the original recommendations (Shriver 2018).

Furthermore, the collaborative recommendation system is sensitive to recommendation attack or shilling attack. In view of this attack, mischievous users enter a large amount of affected profiles toward the system. To differentiate the certain profiles, the affected profiles are known as attack profiles. Recommendation attack is split into nuke and push attack that encourage and discourage the recommended target item. Some common attack models are random, average, bandwagon, love/hate and segment attacks. The durability of shilling attack is mainly measured by filler size and attack size. Some traditional machine learning techniques are used for detecting the shilling attack such as clustering techniques, C4.5, SVM and KNN (Zhou et al. 2020). Table 1 illustrates comparison between some existing recommendation techniques.

The major contributions of this paper are:

1. An insight into the up-to-date shilling attacks in the recommender system.
2. The comparative analysis of unsupervised Machine Learning algorithms in detecting different kinds of Shilling attacks on Netflix dataset.
3. Validation of Machine Learning Algorithms on dataset of 26 K videos on Trending YouTube Video Statistics, to predict the user preferences for a particular genre of videos.

1.1 Role of recommender system in defense systems

In recent days, recommender systems play an important role in various domains such as news, tourism, e-commerce to handle information overload and provide personalized services. At the same time, there are many challenges involved in adapting recommender systems in defense and cyber security systems (Gadepally et al. 2016). Mainly, the defense systems rely on the availability, integrity and confidentiality of the network. However, the performance of recommender systems suffers various cyber attacks and one such attack is shilling attacks (O'Mahony et al. 2005). Attacks in the literature are validated against the CF systems. Identification of these attacks is heterogeneous in nature, and type of the data is an open challenge, e.g., data of images, music, videos, etc. Development of generalized attack model is required for the attacker's knowledge in different applications. Recommendation for the defense systems is required to protect the data timely from the unauthorized users. In this work, we propose a machine learning-based approaches to deal with shilling attacks in recommender systems in defense systems (Deldjoo et al. 2021). Authors have presented attack models on intrusion detection and have worked upon the security models (Mittal et al. 2019,2020; Vinayakumar et al. 2019); however, none of these have considered the shilling attacks which are presented in this paper.

The rest of the article is structured as: Sect. 2 delivers the existing work based on the recommender system among shilling attacks. Section 3 gives an overview of shilling attack and existing algorithms used to detect the shilling attack. Section 4 focuses on the results and discussions on the experimental study conducted to predict and analyze the user preferences on the dataset of YouTube videos. In the end, the article is concluded in Sect. 5.

Table 1 Comparison between some existing Recommender Techniques

Techniques/ Approaches	Benefits	Detriments
Content-Based Filtering	It does not rely on the preferences of another user Comparison between items is possible	To make a valid recommendation is difficult Cold-start problem for new user
Collaborative Filtering	Better results provide when the statistic of customers and the number of ratings are available	Data Sparsity Problem Cold-Start dilemma in case of new item and new user
Knowledge Based	No need of rating No cold-start problem	User Knowledge required Static model
Demographic System	Recommendations are accurate only a particular region are made, thus escape of invalid ones	Relatively less research in this particular technique
Community-Based System	It facilitates simple and extensive data acquisition similar to users social relations	Recommendations are not always accurate

2 Related work

The exposure of shilling attacks in collaborative RS got attention of several researchers in the recent past. There are various disclosure methods that have been suggested up to the previous time that can be divided into semi-supervised method, unsupervised method and supervised methods.

In the league of supervised detection methods, the authors have presented a two-stage method by exploiting a support vector machine (SVM) situated classifier and target item analysis. This approach well identifies the attacks along with a large filler size, and the achievement of detecting the attack is very poor when the detection of attack with small filler and small attack sizes (Cai and Zhang 2019). Similarly, the authors presented several statistics-based metrics for high density attack the low-density attack profiles (Zhou et al. 2020). In another work, the authors presented a fine-grained RS for the social ecosystem, created to recommend published content by user's friends. The main purpose behind is to find the consistent way to obtain information representation of overlapping interests in various sub-categories (Aivazoglou et al. 2020). Furthermore, an ontology-based endorsement RS data used in social networking websites are recommended, and this method is sub-sustained by a mutual ontology model worthwhile to serve as both the contented advertisement and users profiles (García-Sánchez et al. 2020). In the category unsupervised detection model, further the RS is very appropriate for helping customers with the existing features. Recommender systems use several algorithms to access groupings including K-mean algorithm, mini batch K-mean, mean-shift algorithm and clustering algorithm (Putri et al. 2020). In another work, researchers proposed a new method of recommendation hybrid tweet-based recommendation with Latent Dirichlet Allocation (LDA) and generalized some matrix factorization using some supervised learning-based neural network (Goldberg et al. 1992). Similarly, in other work researchers discussed about the shilling attack, some existing attack models, attack detection algorithms and cost–benefit analysis. This is the top up for our understanding of the latest and complete survey around shilling attack (Gunes et al. 2014). Table 2 provides an insight into the existing work based on the recommender system in various application domains. Some of the security attacks are determined and solved with the help of machine learning methods, but they have ignored the shilling attacks (Kumar and Reddy 2014; Bland et al. 2020; Lucia and Cotton 2020; Mao et al. 2020). Undoubtedly, security threats can also be raised due to the false information and to avoid it, experimental analysis is conducted in this paper.

This paper is on the shilling attacks on the RS application Entertainment category where various machine learning algorithms are applied to check the accuracy of the identification of shilling attacks on trending YouTube videos.

3 Attack models

Recommender system helps a user find items or products according to their interest and preferences. When a user enters into a system and starts injecting to a system, it is known as a shilling attack. Attacks contain profiles that include biased information related to mischievous users. Attackers are successful even if they have little to negligible knowledge about the system. Segment-based attacks in contradiction to collaborative filtering guarantee that the

Table 2 Existing work

System	Recommender Technique	Algorithm	Application/Recommends
(Gunes et al. 2014)	Collaborative Filtering	Query Language	Spam Detection
(Shokeen and Rana 2020)	Collaborative Filtering	Cosine Similarity	E-commerce
(Logesh et al. 2019)	Collaborative Filtering	Matrix Factorization	Travel
(Bozanta and Kutlu 2019)	Hybrid Filtering and aggregation	Artificial Neural Network(Weighted)	Venue
(Mizgajski and Morzy 2019)	Hybrid Filtering	Heuristic Similarity (Signal to signal)	Online Content
(Chen et al. 2019)	Collaborative Filtering	PCA and SVM	Malicious Samples
(Missaoui et al. 2019)	Collaborative filtering	Statistical Language model	Tourism and Travel
(Batmaz et al. 2020)	Collaborative Filtering	KNN and SVM	Binary Data (Malicious)
(Kaur and Goel 2016)	Collaborative Filtering	Matrix Factorization	Entertainment

item that is being attacked by the attacker would most probably be suggested to the target users. Further, some common attacks are Average, Bandwagon, random, Segment, Reverse Bandwagon and Love/hate attacks. Average attack desires knowledge around the system, because it considers separate average ratings for each item rather than the universal rating system. Attackers choose items randomly and ratio those using ordinary scattering with mean rating and standard deviation. Random attack is low knowledge level attack. The items are selected at random and are rated using ordinary scattering based on standard deviation (SD) and average valuation of the system. Items set is empty or null. The target item set is rated according to the form of attack like nuke or push attack. The most trending and liked by lots of user's items are popular. Therefore, the chances are high that a high attacker becomes similar to the actual user. So, it is difficult to identify the attacker amidst the actual users. The malicious user makes a collection of segmented objects having higher chance of being desired by target users, who belong to him/her in a certain segment. This creates a huge impact of attack wherein minimum ratings are given to the items that are in the filler set. The Reverse Bandwagon attack is also a type of nuke attack in that preferred things are liked from disliked things, and the maximum rating values are given by the user. The love/hate attack is an efficient nuke attack in which the non-target items randomly choose the filler items. The rating of all filler items is maximal instead of the target item (Cai and Zhang 2019).

Similarly, the stability of shilling attack is measured into two measured: Filler and Attack size. The statistic value of a specific attack profile is known as filler size. The inclusion of ratings has approximately low cost for attackers when the attack profiles are correlated with the cost then create some attack profiles. Furthermore, always customers rate a small fraction. The filler size is set to 1% to 20%. Further, the statistic of fake profiles included into the system is called attack size. The well developer RS is commonly set to 1–15% attack size, as long as shilling attack has a correlation with the cost that depends on the level of creation (Tong et al. (2018)). Table 3 illustrates the Research work done on attack schemes in the recommender system.

3.1 Coverages

There are some distance attributes that are used to calculate the correlation between two probability distributions:

- *Hellinger Distance (HD)* The Hellinger distance is used to compute the correlation is bounded by two possible distributions that totally depend on the Hellinger integral. In terms of measure theory the HD is, When P and Q implies the two probable points that are truly stable with admiration of the third probability distribution. The HD formula written as:

$$H(P,Q)=\frac{1}{\sqrt{2}}\{|\sqrt{p-q}|\} \tag{1}$$

  where $|\sqrt{p}-\sqrt{q}|$ is the difference square root vectors of Euclidean form.
- *Density-Based Spatial Clustering of Applications with Noise* DBSCAN is the ultimate clustering technique, that is situated on clustering and the points with close neighbors are clustered together. The method splits the point's space into core points, outliers and accessible points. All points of the same cluster are mutually associated with each other. Those points that are inaccessible from other points are well-known as outliers. Therefore, there are two constraints required by DBSCAN one is predefined distance and other is minimum number of neighbors. The main benefit of this clustering method is that it does not need to specify the cluster in advance (Si and Li (2020)).

Table 3 Existing work done on various attack schemes

Attacks	Publications
Average Attack	Burke et al. (2005), Hurley et al. (2007)
Random Attack	O'Mahony et al. (2004), Lam and Riedl al.(2004), Burke et al.(2005), Mobasher et al. (2007)
Bandwagon Attack	Burke et al.(2005), O'Mahony et al. (2005), Hurley et al. (2007)
Segment Attack	Burke et al.(2005), Mobasher et al. (2007)
Reverse Bandwagon Attack	Mobasher et al. (2007), Zhang (2009)
Love/Hate Attack	Mobasher et al. (2007), Zhang (2010)

- *K-nearest neighbors (KNN)* KNN is one of the transparent machine learning algorithms based on the supervised machine learning method. It stores the all feasible data and classifies it into the recent data on the behalf of similarity. KNN is also used for regression and classification but most widely used for classification problems. In the training phase, the KNN algorithm just stores the dataset and when it gets new data, then it classifies into new data. The KNN algorithm mainly works on the Euclidean distance that is between the two pints.

$$ED(x, y) = \sqrt{\sum_{i=1}^{n} (qi - pi) * 2} \tag{2}$$

where p and q are represented as Euclidean parameters.

3.2 Attack profile distributions

Our main intention is to review the label of each profile as earlier being part of an attack. The attack profiles can be detected using various classification techniques. For this, a training set is created to train a classifier so that we will be able to differentiate between the mischievous profiles and the generic profiles. There are two types of detection attribute created:

3.2.1 Generic attributes

Generic attributes are made by seeing a profile as a full model specific aspect. We predict the comprehensive statistical trademark of attack profiles desires to fluctuate from that accurate profile. This distinction comes against two origins: the value given to an objective item and distribution of the rating among filler items. Some generic attributes are: RDMA, WDMA, DegSim, LengthVar (Rating deviation from mean agreement, Weighted Deviation from Mean Agreement, Degree of similarity with Top neighbors and Length Variance, respectively).

- *RDMA*: RDMA was designed to analyze attackers over the average variance per time, filled by the other statistic of value for that item.
- *WDMA*: WDMA is toughly dependent on RDMA, however, weighs on higher rating deviation being sparse items. To provide higher information gain, we have found variants.
- *DegSim*: The DegSim aspect is established on the average interaction of K-nearest neighbor's profile. The correlation between users is determined by Pearson's correlation. When the statistics of input points are small, familiar characteristics are correlation-based measures. Since the total things co-rated by two customers that resolve their similarity. When we calculate the DegSim, the producing attribute is 'DegSim.'
- *LengthVar*: It defines how much the size of the known profile differs from the ordinary length in the table. If the possible items in a large number, unlikely the huge amount of profiles comes from actual users, as opposed to soft-bot implements a profile injection attack. However, this aspect is mainly efficient at identifying attacks with big filler size.

3.2.2 Model specific attributes

For finding the specific feature of attack model, model specific attribute is used. Although the profiles are slight, consist of fewer filler items is especially true. These attacks can be successful, and we pursue the generic attributes with some designed characteristics of our attack models. The main aim of model specific aspects is to make the special signature of our particular attack model. The detection of attributes is used to rate the statistical features. The detection model detects the segment of each and every profile that augmented its correlation to the attack model. Some model specific attributes are: Mean variance—for detection of average attack and Target Focus detection model.

- *Mean Variance for detection of average attack model*: Its profile splits into three portions: extreme value specified to the target item, filler item given other rating and unrated items. This model approximately selects the objective of items and other valuable items converted into fillers. With the explanation of average attack, the value of filler will be popular such as they jointly compete the filler items with the average value. With the average attack, we assume that profile will grow and would advertise a huge degree of correlation amongst its values and the single item chosen as a target item.

- *Target Focus Detection model*: These attributes have robust on inter profile data, target focus, and, however, robust on intra-profile data. It is beneficial to review the solidity of target items over profiles. The main advantage of this is the partitioning correlates with a model-based attribute that defines the set of imagined targets to identify each profile (Si and Li 2020).

3.2.3 Measurement of performance parameters

For measuring the performance of the classifiers, Recall, Precision and F-measure parameters are used and explained below:

- *Recall*: Recall is the rate of correctly concluded positive observations in actual class. Recall for the values the data (from 0 to ith) of the present article is computed by Eq. 3.

$$Recall = \frac{TruePositive_0^i}{TruePositive_0^i + FalseNegative_0^i} \quad (3)$$

- *Precision*: The quantity of true positive partitioned by the number of true positives in addition to the number of false positives.

$$Precision = \frac{TruePositive_0^i}{TruePositive_0^i + FalsePositive_0^i} \quad (4)$$

- *F-measure*: It is the weighted-average of recall and precision. Thus, this takes both false negative and false positive into account. F-measure is also called F1-score or F-score.

$$F - measure = 2 * \frac{(Recall * Precision)}{(Recall + Precision)} \quad (5)$$

There are various metrics that have been proposed for evaluating the effective shilling attack detection schemes. Table 4 illustrates the accuracy metrics.

3.3 Methodology

Methodology used in the article is observed from Fig. 1.

- Data Set Selection: Based on the literature review, we identified and selected the most suitable dataset for the validation of machine learning algorithms in shilling attacks. The collective dataset is on Trending YouTube Video Statistics from Kaggle updated ten years ago.
- Data preprocessing: In this phase, we preprocessed the dataset using cleaning or filling the missing values with the correlation. For this, data regarding service domain sites need to be considered. Therefore, we analyzed and categorized a data set as per requirements of study.
- Identification of Push and Nuke Attack: There are some existing classification techniques which are used for identification or predicting the value of attack. We implemented existing classification techniques on the selected data set for identification of Shilling attacks.
- Performance analysis: We conducted performance analysis based on the results obtained from the performance analysis phase (subsequent section).

3.4 Comparative observations of unsupervised methods under various sizes of attacks for different filler sizes

In this section, various algorithms are analyzed under the different categories of shilling attacks. The most popular datasets considered for this study are Netflix and MovieLens, which gives the high data volume. A comparative analysis of existing Machine learning algorithms for detecting Average Attack, Bandwagon Attack, Random Attack over the 3%, 5%, 8% and 10% filler size are given in Table 5 to Table 13 for Netflix dataset.

As shown in Fig. 2, the precision over average attack, when the 3% of filler size across 3%, 5%, 8%, and 10% attacks size, the precision of PCA-VarSelect is 57%,66%, 75% and 74%, the precision of CBS is 53%, 68%, 83% and 85%, the precision of EUB-DAR is 30%, 44%, 55% and 60%, the precision of UD-HMM is 75%, 75%, 84% and 94% and the precision of DSA-AURB is 94%, 94%, 99% and 97%. According to obtained results, DSA-AURB performs very well with a 3% filler size with a 10% attack size gives 97% detection of shilling attack for average attack. EUB-DAR cannot be filtered by most of the genuine users.

According to Fig. 3, we conclude the precision under the Random attack, when the 3% of filler size across 3%, 5%, 8% and 10% attacks size, the precision of PCA-VarSelect is 56%,68%, 76% and 79%, the precision of CBS is 62%, 63%, 82% and 85%, the precision of EUB-DAR is 30%, 33%, 49% and 60%, the precision of UD-HMM is 72%, 76%, 92% and 94% and the precision of DSA-AURB is 92%, 98%, 99% and 99% corresponding to attack sizes. After the implementation, DSA-AURB performs very well with a 3% filler size with 8% and 10% attack size gives 99% detection of shilling attack for

Table 4 Performance Evaluation Metrics (Tong et al. 2018)

	Evaluation of metrics
Evaluating detection methods	Recall, precision, correctness, ROC, specificity, detection rate and F1-measure
Assessing shilling effects	Average hit ratio, average prediction shift, high rating ratio, precision shift
Evaluating robust algorithm	Mean absolute error, normalized absolute error, root mean squared error

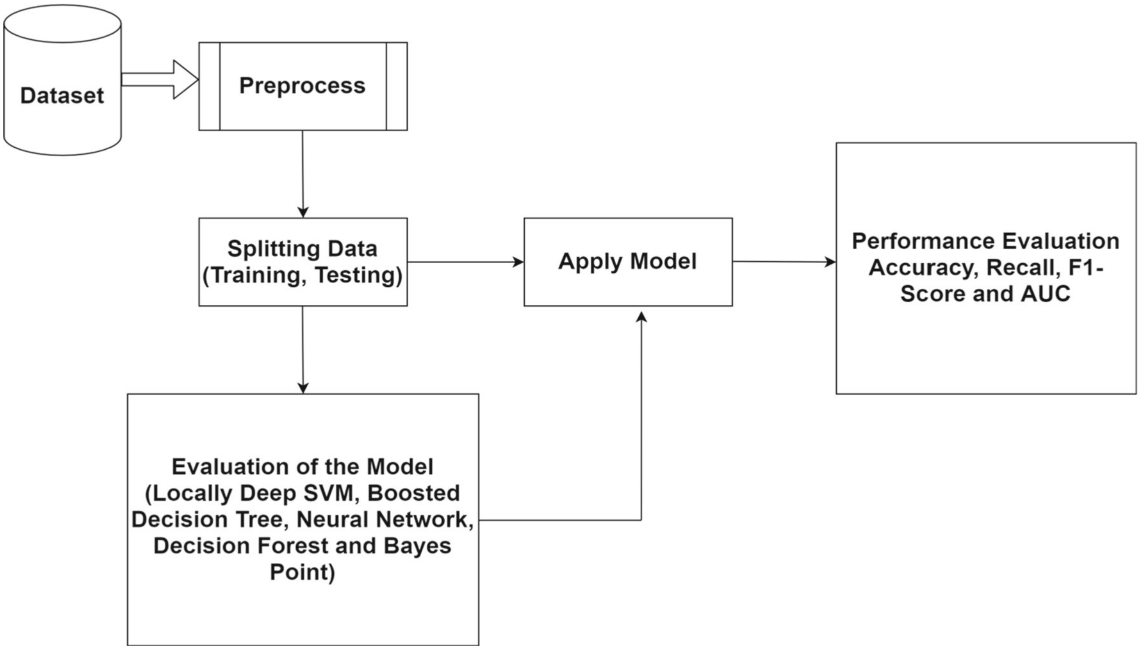
Dataset
Preprocess
Splitting Data (Training, Testing)
Apply Model
Performance Evaluation Accuracy, Recall, F1-Score and AUC
Evaluation of the Model (Locally Deep SVM, Boosted Decision Tree, Neural Network, Decision Forest and Bayes Point)

Fig. 1 Proposed methodology

Table 5 Comparison of Precision over Average Attack

	3% Indegree			
	3%	5%	8%	10%
PCA-VarSelect	0.5738	0.6667	0.7512	0.7477
CBS	0.5357	0.6875	0.838	0.8557
EUB-DAR	0.3089	0.4411	0.5577	0.6088
UD-HMM	0.7552	0.7542	0.8467	0.9482
DSA-AURB	0.9481	0.94	0.995	0.9707

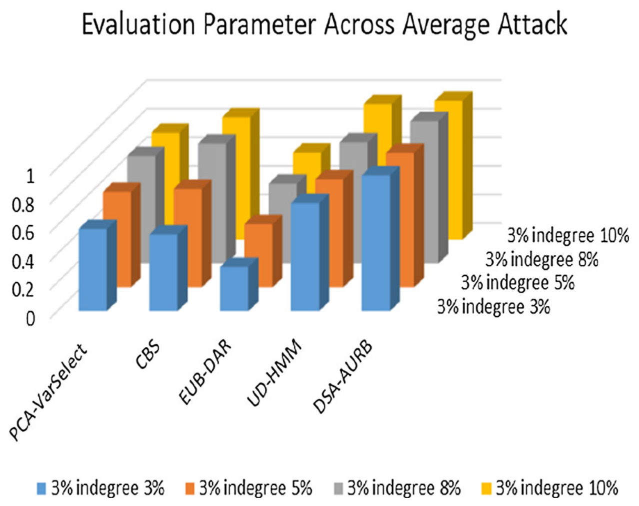

Fig. 2 Comparative analysis of multiple approaches in detecting Average Attack

random attack. EUB-DAR cannot be clarified for most of the unaffected users.

From Fig. 4, we analysis the precision under the Bandwagon attack, when the 3% of filler size across 3%, 5%, 8%, and 10% attack size, the precision of PCA-VarSelect is 0%,8%, 27% and 38%, the precision of CBS is 57%, 70%, 84% and 87%, the precision of EUB-DAR is 35%, 47%, 59% and 67%, the precision of UD-HMM is 73%, 74%, 75% and 86% and the precision of DSA-AURB is 96%, 98%, 99% and 99% as, respectively, attack sizes. According to the obtained results, DSA-AURB performs

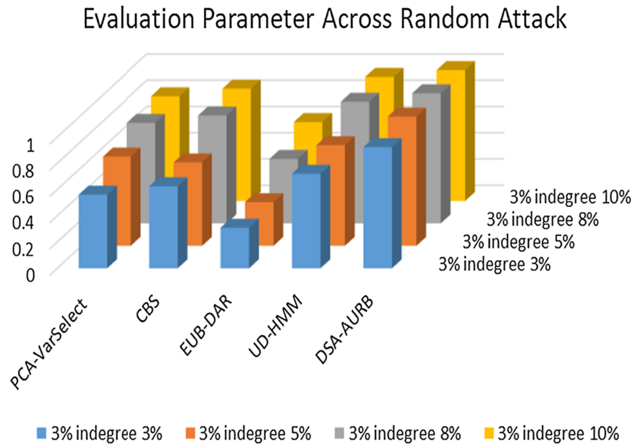

Fig. 3 Comparative analysis of multiple approaches in detecting Random Attack

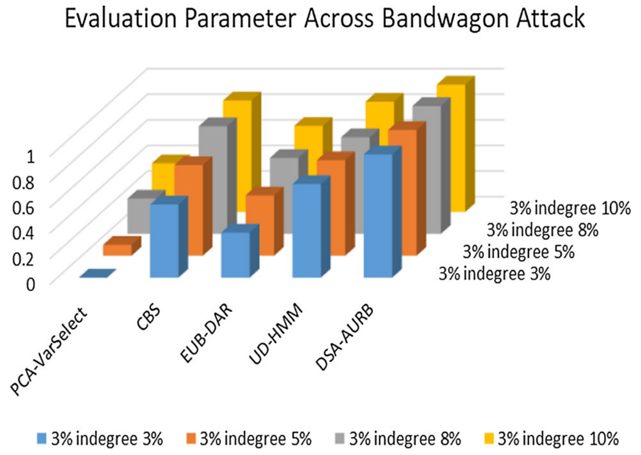

Fig. 4 Comparative analysis of multiple approaches in detecting Bandwagon Attack

very well with a 3% filler size with a 8%, 10% attack size gives 99% detection of shilling attack for average attack. EUB-DAR cannot be clarified for most of the unaffected users.

From Fig. 5, the recall over the Average attack, when the 3% of filler size across 3%, 5%, 8%, and 10% attack size can be measured and the recall of PCA-VarSelect is 58%,71%, 80% and 87%, the recall of CBS is 54%, 72%, 89% and 95%, the recall of EUB-DAR is 97%, 98%, 97% and 92%, the recall of UD-HMM is 98%, 99%, 96% and 100% and the recall of DSA-AURB is 99%, 100%, 99% and 99% as, respectively, attack sizes. According to the obtained results, DSA-AURB performs very well with a 3% filler size with 8% and 10% attack sizes and gives 99% detection of shilling attack for average attack. In random attack, the CBS and VarSelect cannot detect the genuine profiles on the 3% filler size and 3% attack size according to other methods.

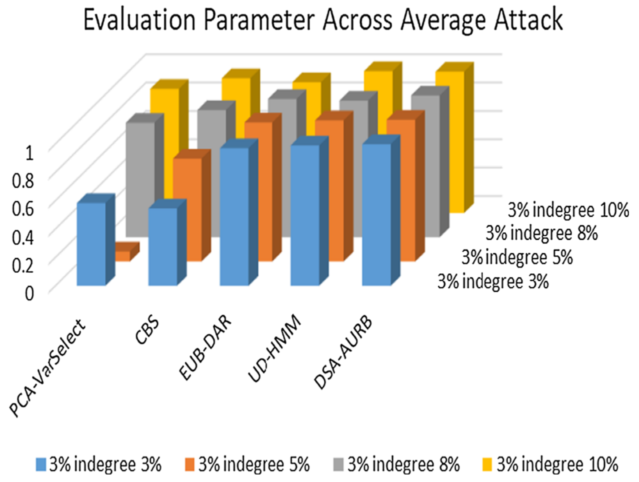

Fig. 5 Comparative analysis of multiple approaches in detecting Average Attack

From Fig. 6, let us take the recall under the Random attack, when the 3% of filler size across 3%, 5%, 8% and 10% attack size, the recall of PCA-VarSelect is 57%,71%, 82% and 87%, the recall of CBS is 63%, 67%, 88% and 95%, the recall of EUB-DAR is 93%, 97%, 99% and 99%, the recall of UD-HMM is 97%, 99%, 100% and 99% and the recall of DSA-AURB is 99%, 99%, 99% and 99% as, respectively, attack sizes. According to the obtained results, DSA-AURB performs very well with a 3% filler size with a 3%. 5%, 8%, 10% attack size gives 99% detection of shilling attack for average attack. In random attack, the PCA-VarSelect cannot detect the genuine profiles on the 3% filler size and 3% attack size according to other methods.

From Fig. 7, we conclude the recall under the Bandwagon attack, when the 3% of filler size across 3%, 5%, 8% and 10% attack size, the recall of PCA-VarSelect is 0%, 0%, 29% and 41%, the recall of CBS is 58%, 74%, 91% and 96%, the recall of EUB-DAR is 96%, 99%, 95%

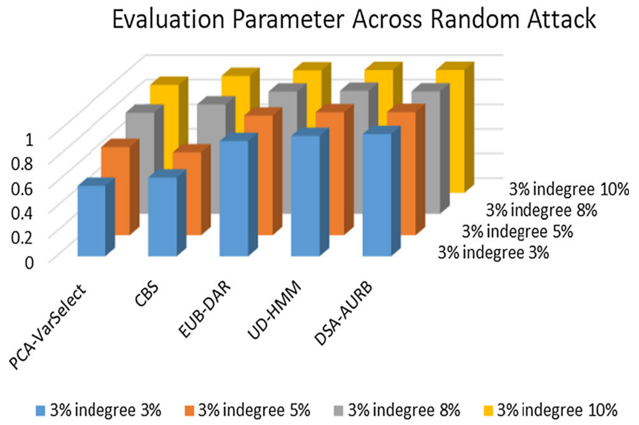

Fig. 6 Comparative analysis of multiple approaches in detecting Random Attack

and 95%, the recall of UD-HMM is 99%, 98%, 99% and 99% and the recall of DSA-AURB is 99%, 99%, 99% and 99% as, respectively, attack sizes. According to the obtained results, DSA-AURB and UD-HMM perform very well with a 3% filler size with a 3%. 5%, 8%, 10% attack size gives 99% detection of shilling attack for average attack. In random attack, the PCA-VarSelect cannot detect the genuine profiles on the 3% filler size and 3%, 5%, 8% and 10% attack size according to other methods.

From Fig. 8, we analyze the F1-measure under the Average attack, when the 3% of filler size across 3%, 5%, 8% and 10% attack size, the F1-measure of PCA-VarSelect is 57%,68%, 77% and 80%, the F1-measure of CBS is 54%, 70%, 86% and 90%, the F1-measure of EUB-DAR is 46%, 60%, 70% and 73%, the F1-measure of UD-HMM is 85%, 85%, 90% and 97% and the F1-measure of DSA-AURB is 97%, 96%, 99% and 99% as, respectively, attack sizes. After the implementation, DSA-AURB and UD-HMM perform very well with a 3% filler size with a 10% attack size gives 99% and 97%, respectively, detection of shilling attack for average attack. In Average attack, the EUB-DAR cannot detect the genuine profiles on the 3% filler size and 3% attack size according to other methods.

From Fig. 9, we observe the F1-measure under the Random attack, when the 3% of filler size across 3%, 5%, 8% and 10% attack size, the F1-measure of PCA-VarSelect is 56%,69%, 79% and 83%, the F1-measure of CBS is 63%, 65%, 85% and 90%, the F1-measure of EUB-DAR is 46%, 49%, 65% and 74%, the F1-measure of UD-HMM is 82%, 86%, 96% and 97% and the recall of DSA-AURB is 95%, 99%, 99% and 99% as, respectively, attack sizes. According to the experimental results, DSA-AURB performs very well with a 3% filler size with a 5%, 8%, 10% attack size gives 99% detection of shilling attack for average attack. In random attack, the EUB-DAR cannot detect the genuine profiles on the 3% filler size and 3%, 5% attack size according to other methods.

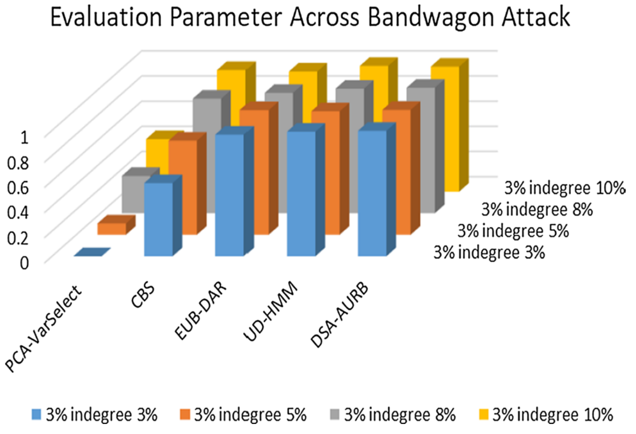

Fig. 7 Comparative analysis of multiple approaches in detecting Bandwagon Attack

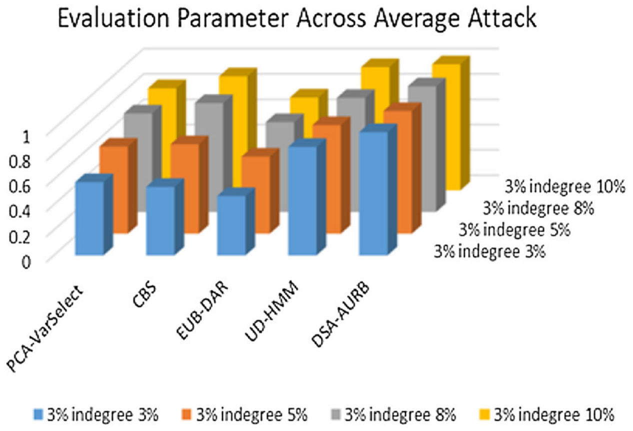

Fig. 8 Comparative analysis of multiple approaches in detecting Average Attack

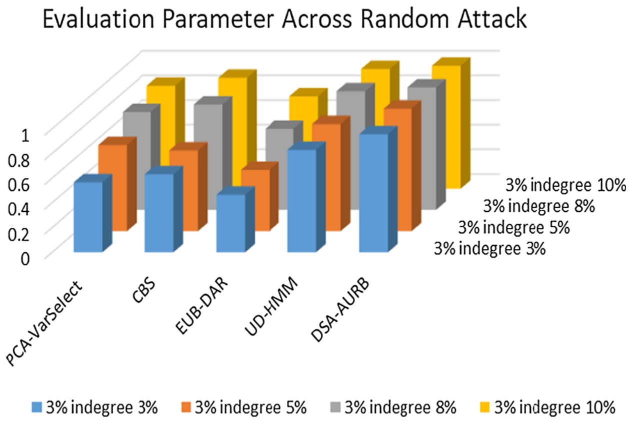

Fig. 9 Comparative analysis of multiple approaches in detecting Random Attack

According to experimental results, the F1-measure under the Bandwagon Attack (Fig. 10), when the 3% of filler size across 3%, 5%, 8% and 10% attack size, the F1-measure of PCA-VarSelect is 0%,0%, 28% and 39%, the F1-measure of CBS is 57%, 72%, 87% and 91%, the F1-measure of EUB-DAR is 51%, 63%, 73% and 78%, the F1-measure of UD-HMM is 84%, 84%, 85% and 92% and the F1-measure of DSA-AURB is 97%, 98%, 99% and 99% as, respectively, attack sizes. According to the experimental results, DSA-AURB performs very well with a 3% filler size with a 8%, 10% attack size gives 99% detection of shilling attack for average attack. In Bandwagon attack, the PCA-VarSelect cannot detect the genuine profiles on the 3% filler size and 3%, 5%, 8% and 10% attack size according to other methods.

From Tables 5, 6, 7, 8, 9, 10, 11, 12 and 13, Precision, Recall, F1-measure for the five unsupervised methods such as PCA-VarSelect, CBS, EUB-DAR, UD-HMM, DSA-

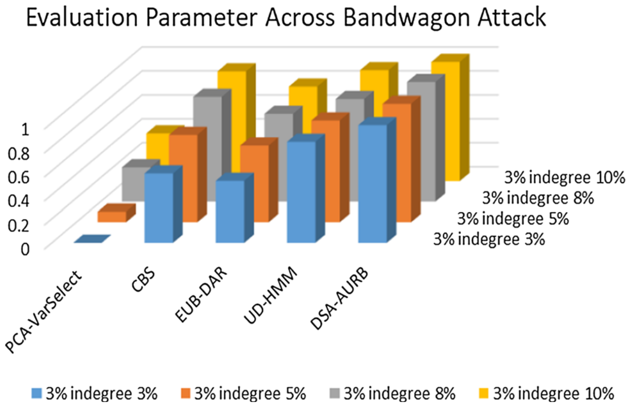

Fig. 10 Comparative analysis of multiple approaches in detecting Bandwagon Attack

AURB can be analyzed under various attacks including Average, Random and Bandwagon attack with 3% of filler size and 3%, 5%, 8% and 10% of attack sizes on Netflix dataset, respectively. The concluded results indicate the DSA-AURB is more supportive for detecting the recommender system against shilling attacks.

3.5 Unsupervised detection methods

To demonstrate the performance of the expected methods, DSA-AURB is compared with the following methods.

- *PCA- VarSelect*: It is an unsupervised shilling attack disclosure method that is used for good performance detection, if the statistic of attack users is already knowledgeable. In the analysis, we consider the size of attack is familiar in advance.
- *CBS*: The Catch the black sheep is a cooperative outline for detecting the shilling attack based on fake action propagation, which concludes the spam possibilities for each item and user. Before implementing CBS, we need the prior information for CBS. For the experiment analysis, we aimlessly choose 10 attack users from the classified attackers as the spam users.

Table 6 Comparison of Precision over Random Attack

	3% Indegree			
	3%	5%	8%	10%
PCA-VarSelect	0.5607	0.68	0.7651	0.7973
CBS	0.625	0.6354	0.8217	0.8557
EUB-DAR	0.3091	0.3322	0.4907	0.6012
UD-HMM	0.72	0.7667	0.9269	0.9452
DSA-AURB	0.9231	0.9847	0.9913	0.997

Table 7 Comparison of precision over bandwagon attack

	3% Indegree			
	3%	5%	8%	10%
PCA-VarSelect	0	0.0857	0.2756	0.38
CBS	0.5714	0.7083	0.8408	0.8706
EUB-DAR	0.3515	0.4714	0.593	0.6725
UD-HMM	0.7308	0.7466	0.7551	0.8616
DSA-AURB	0.9629	0.9825	0.9988	0.995

Table 8 Comparison of recall over average attack

	3% Indegree			
	3%	5%	8%	10%
PCA-VarSelect	0.5831	0.071	0.8075	0.875
CBS	0.5455	0.7253	0.8966	0.9503
EUB-DAR	0.9701	0.98	0.9752	0.9221
UD-HMM	0.9889	0.995	0.9635	1
DSA-AURB	0.9983	1	0.9994	0.9965

Table 9 Comparison of recall over random attack

	3% Indegree			
	3%	5%	8%	10%
PCA-VarSelect	0.57	0.714	0.8225	0.877
CBS	0.6364	0.6703	0.8897	0.9503
EUB-DAR	0.9333	0.97	0.9938	0.995
UD-HMM	0.975	0.9967	1	0.9983
DSA-AURB	0.99	0.998	0.995	0.999

Table 10 Comparison of recall over bandwagon attack

	3% Indegree			
	3%	5%	8%	10%
PCA-VarSelect	0	0.09	0.2963	0.418
CBS	0.5818	0.7473	0.9103	0.9669
EUB-DAR	0.9667	0.99	0.9563	0.955
UD-HMM	0.99	0.981	0.99	0.9998
DSA-AURB	0.9967	0.992	0.9975	0.994

- *EUB-DAR*: The estimating user's behavior toward detected anomalous ratings in rating system is an unsupervised shilling attack detection method, which is used for detecting the attackers by analysis of target

Table 11 Comparison of F1-measure over average attack

	3% Indegree			
	3%	5%	8%	10%
PCA-VarSelect	0.5784	0.6877	0.7783	0.8064
CBS	0.5406	0.7059	0.8609	0.9005
EUB-DAR	0.4686	0.6084	0.7096	0.7334
UD-HMM	0.8564	0.858	0.9013	0.9734
DSA-AURB	0.9726	0.9691	0.9924	0.9957

Table 12 Comparison of F1-measure over random attack

	3% Indegree			
	3%	5%	8%	10%
PCA-VarSelect	0.5653	0.6966	0.7928	0.8353
CBS	0.6306	0.6524	0.8543	0.9005
EUB-DAR	0.4644	0.4949	0.657	0.7495
UD-HMM	0.8283	0.8667	0.9621	0.971
DSA-AURB	0.9554	0.9913	0.9931	0.998

Table 13 Comparison of F1-measure over bandwagon attack

	3% Indegree			
	3%	5%	8%	10%
PCA-VarSelect	0	0.0878	0.2856	0.3981
CBS	0.5766	0.7273	0.8742	0.9162
EUB-DAR	0.5155	0.6387	0.7321	0.7892
UD-HMM	0.8409	0.8479	0.8567	0.9256
DSA-AURB	0.9795	0.9872	0.9981	0.9945

items. Similarly, in the graph topological structure of attack users is detected on the basis of similarity.

- *UD-HMM*: The Hidden Markov model and Hierarchical clustering method are an unsupervised shilling attack detection method, which usually performs excellent performance for detecting various attacks.

4 Result analysis and discussions

In this section, we have evaluated and analyzed the performance of various machine learning algorithms. We performed an experimental study on data of 26 K strategy videos on the Trending YouTube Video Statistics, collected on 10 months ago, using the YouTube history. Using this dataset, insights can be gained into the user preferences for a particular genre of history or to predict the success of videos based on the number of ratings, etc.

We analyze and evaluate various machine learning algorithms namely: Boosted decision forest, Boosted decision tree, neural network, and Deep decision and Bayes point. The evaluation of the algorithm is done based on Precision, Accuracy, Recall, F1-measure and Area under curve (AUC) on Microsoft Azure. Table 14 illustrates the results.

From Table 14, it can be observed that Boosted Decision Tree has performed really well with 99% of accuracy and AUC of 100% on 80% percent of train models and 20% of testing models. Moreover, it can further be observed that both that the accuracy and AUC of average perception are minimum compared with all other algorithms. Figure 10 illustrates and compares the performance of the used algorithms on the basis of precision, accuracy, recall, f1-measure and AUC evaluated for the video's dataset.

After concluding the results (Fig. 11), we are focusing on another parameter that is time taken by Machine learning algorithms on results. Table 15 shows the different time taken by different algorithms that are used for concluding the results. In terms of time, the Bayes Point takes very long time and Boosted Decision Tree takes very less in comparison with other algorithms.

It can be observed that the Boosted Decision tree has performed really well with 33% less time and Bayes Point takes a maximum time with the 50% in comparison with other algorithms of the YouTube video's dataset. Therefore, the Boosted Decision tree gives the best performance according to this dataset. Figure 12 illustrates the processing time variations on the basis of Decision Forest, Boosted Decision Tree, Neural Network, Deep Decision and Bayes Point evaluated for video dataset.

In the present system, it is observed that DSA-AURB works well for the large data set. However, this algorithm

Table 14 Comparison of machine learning algorithms on YouTube trending videos

Algorithm	Precision	Accuracy	Recall	F1-Measure	AUC
Decision Forest	0.993	0.993	1	0.996	0.724
Boosted Decision Tree	1	0.999	0.999	0.999	1
Neural Network	1	0.998	0.998	0.999	1
Deep Decision	0.999	0.999	1	1	1
Bayes Point	0.998	0.998	1	0.999	0.885

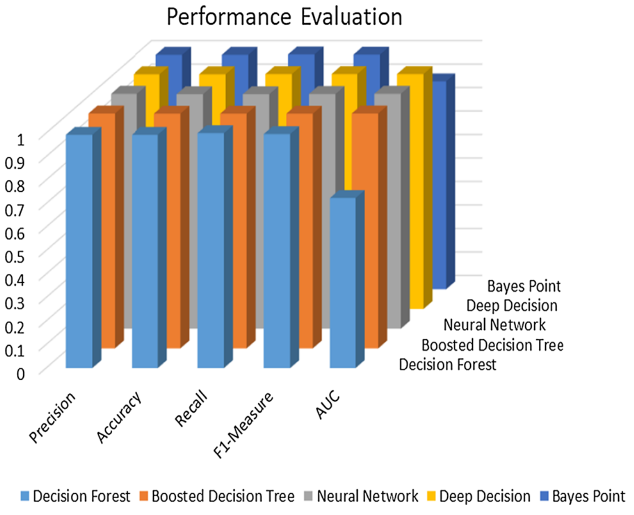

Fig. 11 Comparative of Bayes Point, Boosted Tree, Decision Forest and Neural Network Algorithms

Table 15 The time variations on Machine Learning Algorithms

Algorithm	Timestamp
Decision Forest	25
Boosted Decision Tree	20
Neural Network	22
Deep Decision	28
Bayes Point	30

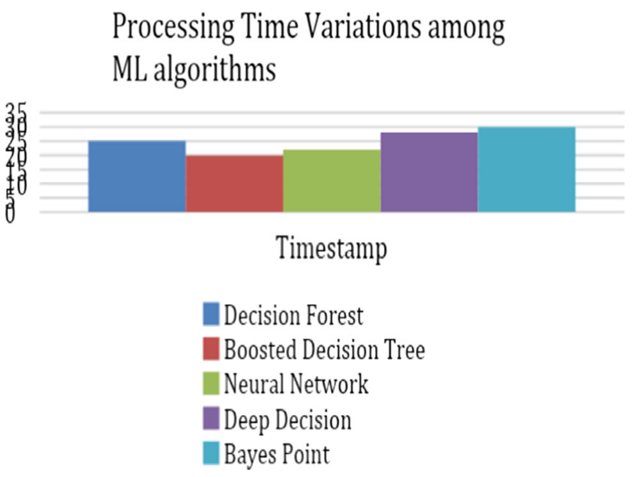

Fig. 12 Processing Time Variations for Machine Learning Algorithms

has given the accuracy for the large dataset and it is not recommendable for the small dataset. This is the research gap in the present article. A new model is required to cope up with both small and large dataset.

Table 16 Parameters of YouTube data set

Sr No	Name	Type	Description
1	vedio_id	string	Video ID
2	trending_date	integer	Trending Date
3	title	string	Title
4	channel_title	string	Channel Title
5	category_id	integer	Category ID
6	publish_time	float	Publish Time
7	tags	text	Tags
8	views	integer	Views
9	likes	integer	Likes
10	dislikes	integer	Dislikes
11	comment_count	integer	Count of comments
12	tubnail_link	text	Tubnail Link
13	comments_disabled	boolean	Disabled Comments
14	rating_disabled	boolean	Disabled Rating
15	vedio_error_or_removed	boolean	Video Error or Removed

5 Conclusion

In this paper, the recommender system along with an introduction of shilling attacks is presented. These attacks are harmful for the realistic application scenarios and can lead to the security endangerments. The implementation and experimental analysis over the unsupervised machine learning algorithms named as PCA-VarSelect, EUB-DAR, DSA-AURB, UD-HMM and CBS with the different filler sizes along with different attack sizes, i.e., 3% filler size in degrees of 3%, 5%, 8% and 10%, respectively, is shown for precision, recall, AUC and accuracy on the Netflix dataset. Performance of DSA-AURB is validated over other algorithms. However, EUB-DAR's performance is less in the identification of the Average Attack, Bandwagon Attack and Random Attack.

Implementation of machine learning algorithms on data of 2 K videos on the YouTube Trending Videos Statistics proves that decision forest is not suitable for identification of shilling attacks. The parameters for the same are observed in Table 16. The study was further aimed toward predicting user preferences for videos using Machine Learning algorithms. Based on the results, it is observed that the Boosted Decision Tree algorithm performed best with an accuracy of 99 percent. All experiments are executed on Microsoft Azure with Intel i3-3240, CPU and 8 GB of memory. In Future, we aim to use hybrid Machine Learning algorithms to predict and classify the shilling attacks.

Funding This study does not have any funding source.

Compliance with ethical standards

Conflict of Interest Authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants and animals performed by any of the authors.

References

Aivazoglou M, Roussos AO, Margaris D, Vassilakis C, Ioannidis S, Polakis J, Spiliotopoulos D (2020) A fine-grained social network recommender system. Soc Netw Anal Min 10(1):8

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

Bilge A, Ozdemir Z, Polat H (2014) A novel shilling attack detection method. Procedia Comput Sci 31:165–174

Bland JA, Petty MD, Whitaker TS, Maxwell KP, Cantrell WA (2020) Machine learning cyberattack and defense strategies. Comput Secur 92:101738

Bozanta A, Kutlu B (2019) HybRecSys: Content-based contextual hybrid venue recommender system. J Inf Sci 45(2):212–226

Cai H, Zhang F (2019) Detecting shilling attacks in recommender systems based on analysis of user rating behavior. Knowl-Based Syst 177:22–43

Chen K, Chan PP, Zhang F, Li Q (2019) Shilling attack based on item popularity and rated item correlation against collaborative filtering. Int J Mach Learn Cybern 10(7):1833–1845

De Lucia MJ, & Cotton C (2020) A network security classifier defense: against adversarial machine learning attacks. In: Proceedings of the 2nd ACM workshop on wireless security and machine learning, pp 67–73

Deldjoo Y, Di Noia T, Merra FA (2021) A survey on adversarial recommender systems: from attack/defense strategies to generative adversarial networks. ACM Comput Surv. https://doi.org/10.1145/3439729

Gadepally VN, Hancock BJ, Greenfield KB, Campbell JP, Campbell WM, Reuther AI (2016) Recommender systems for the department of defense and the intelligence community. Lincoln Lab J, 22(1). https://www.ll.mit.edu/sites/default/files/page/doc/2018-05/22_1_6_Gadepally.pdf

García-Sánchez F, Colomo-Palacios R, Valencia-García R (2020) A social-semantic recommender system for advertisements. Inf Process Manage 57(2):102153

Goldberg D, Nichols D, Oki BM, Terry D (1992) Using collaborative filtering to weave an information tapestry. Commun ACM 35(12):61–71

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

Kaur P, Goel S (2016) Shilling attack models in the recommender system. In: 2016 International conference on inventive computation technologies (ICICT), Vol 2, pp 1–5, IEEE.

Kumar PV, Reddy VR (2014) A survey on recommender systems (RSS) and its applications. Int J Innov Res Comput Commun Eng 2(8):5254–5260

Logesh R, Subramaniyaswamy V, Vijayakumar V, Li X (2019) Efficient user profiling based intelligent travel recommender system for individual and group of users. Mobile Netw Appl 24(3):1018–1033

Mao Y, Huang W, Zhong H, Wang Y, Qin H, Guo Y, Huang D (2020) Detecting quantum attacks: a machine learning based defense strategy for practical continuous-variable quantum key distribution. New J Phys 22(8):083073

Missaoui S, Kassem F, Viviani M, Agostini A, Faiz R, Pasi G (2019) LOOKER: a mobile, personalized recommender system in the tourism domain based on social media user-generated content. Pers Ubiquit Comput 23(2):181–197

Mittal M, Siriaraya P, Lee C, Kawai Y, Yoshikawa T, Shimojo S (2019) Accurate spatial mapping of social media with physical locations. IEEE BSD, Big Data, Los Angeles, USA, pp 9–12 Dec 2019.

Mittal M, Iwendi C, Khanand S, Javed AR (2020) Analysis of security and energy efficiency for shortest route discovery in leach protocol using levenberg-marquardt neural network and gated recurrent unit for IDS. ETT, Wiley.

Mizgajski J, Morzy M (2019) Affective recommender systems in the online news industry: how emotions influence reading choices. User Model User-Adap Inter 29(2):345–379

O'Mahony MP, Hurley NJ, & Silvestre GC (2005) Recommender systems: Attack types and strategies. In AAAI, pp. 334–339.

Putri DCG, Leu JS, Seda P (2020) Design of an unsupervised machine learning-based movie recommender system. Symmetry 12(2):185

Shokeen J, Rana C (2020) A study on features of social recommender systems. Artif Intell Rev 53:965–988

Shriver D (2018) Assessing the quality and stability of recommender systems. University of Nebraska, Lincoln

Si M, Li Q (2020) Shilling attacks against collaborative recommender systems: a review. Artif Intell Rev 53(1):291–319

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

Vinayakumar R, Alazab M, Soman KP, Poornachandran P, Al-Nemrat A, Venkatraman S (2019) Deep learning approach for intelligent intrusion detection system. IEEE Access 7:41525–41550

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