Proceedings of the 2018 International Conference on Machine Learning and Cybernetics, Chengdu, China, 15-18 July, 2018

UNSUPERVISED SHILLING ATTACK DETECTION MODEL BASED ON RATED ITEM CORRELATION ANALYSIS

KEKE CHEN[1], PATRICK P. K. CHAN[1], DANIEL S. YEUNG[2]

[1]School of Computer Science and Engineering, South China University of Technology, Guangzhou, China
[2]Systems, Man, and Cybernetics Society, United States
E-MAIL: kkechen@qq.com, patrickchan@ieee.org, danyeung@ieee.org

Abstract:

Collaborative filtering is one of the most effective methods to deal with the information overload problem by providing a suitable recommendation to users. Recent research indicates that collaborative filtering is vulnerable to shilling attack which aims at manipulating the recommendation result through injecting malicious profiles. Our previous study suggests that applying the rated item correlation to supervised learning increases the accuracy of shilling attack detection. However, label information collection is one drawback of the supervised learning. In this study, an unsupervised detection based on the rated item correlation analysis is devised. The influence of the parameters on the detection accuracy is also discussed. The experimental results demonstrate that our proposed unsupervised detection model achieve a satisfying performance in shilling detection in MovieLens 100K dataset.

Keywords:

Recommender systems; Collaborative filtering; Shilling attack; Rated item correlation

1. Introduction

Recommender systems aim to provide suitable suggestions to users according to their preferences. In the era of big data, recommender systems play an important role as they are able to efficiently identify the potentially interesting products for users. Collaborative filtering (CF), which is one of the most successful and popular recommendation methods [1], has been widely applied in different applications. CF recommends an item based on the user-item dataset, and achieve an accurate result in previous studies.

However, many studies reveal that CF is vulnerable to shilling attacks (or profile injection attacks) [2]. Shilling attack injects a number of bogus user profiles into the recommender system dataset aiming to manipulate the chance of recommending a target item. As the traditional recommender systems do not concern the existence of an adversary, the quality of their recommendation drops significantly [3]. Therefore, it is crucial to consider the robustness of recommender systems in an adversarial environment.

One of the defense methods is data sanitization. Suspected user profiles are identified and removed. Our previous research [4] has shown real and malicious users can be distinguished effectively by considering the rated item correlation in the supervised learning framework. Attack profiles are usually generated by randomly choosing a rated item from a fixed item set in most existing shilling attack [5]. This random selection is different from the behavior of the genuine users who have their preference and tend to rate some items with larger correlation. As a result, malicious users can be identified in the space of the rated item correlation. The experimental results confirm that our model outperforms the popular methods including the supervised detection method which used the classical detection features and the rated item correlation features.

One drawback of our previous model is that label information is required for training samples. As a collection of a labeled sample is costly, unsupervised learning methods are sometimes more preferable than the supervised one. Therefore, we extend the concept of the rated item correlation to unsupervised learning. In this paper, we first analyze how the parameters of the rated item correlation, *i.e.,* the considered interval of the distribution of rated item correlation, on the detection accuracy of each different type of attack. An unsupervised detection model identifying malicious profiles using the rated item correlation is proposed. Experimental results show that the different malicious users can be recognized by our proposed method accurately.

The rest of the paper is arranged as follows. Section 2 intro-

978-1-5386-5214-5/18/$31.00 ©2018 IEEE

duces the concepts that is related to this study. The equation of rated item correlation measurement and the algorithm of the unsupervised detection model are shown in Section 3. In Section 4, the setting of the experiments and the analysis of the experimental results are discussed. At last, the conclusion and future are illustrated in 5.

2. Background

In this section, the concepts of the shilling attack and the existing shilling attack detection models are introduced.

2.1. Shilling Attack Models

Shilling attack aims to change the recommendation frequency of a target item by injecting a set of bogus user profiles which contain a series of fake ratings into a recommender system. Shilling attacks can be categorized as push and nuke attack based on the intention. In *push* (*nuke* attack, an attacker increases (decreases) the recommendation frequency of a target item [6]. When generating malicious profiles, *attack size* (AS) and *filler size* (FS) are considered. AS is the proportion of generated malicious injected into the recommender system, and FS is a proportion of the rated items in each generated malicious profile.

Items in a malicious profile can be separated into four types, *i.e.,* the unrated items (I_u), the filler items (I_f), the selected items (I_s) and the target items (I_t). I_u represents those items which have not been rated in a malicious profile. I_f means the rated items chosen randomly from the whole dataset. I_f increases the number of rated items in a malicious profile, which is different from I_f and I_s represent the items selected by the attacker to increase the similarity between the malicious and genuine profiles. I_t denotes the target items.

Four typical attack models are used as a benchmark in this paper. All the rated items except the target items in *random attack* and *average attack* [7] are I_f, *i.e.,* all the rated items are randomly selected from the whole dataset. However, *Random* attack rates the chosen items randomly from the normal distribution with the mean and standard deviation calculated from all items in the database, while average value is assigned to the chosen items in *average* attack. *Bandwagon attack* [8] considers both of the I_s and I_f, and the I_s contains some popular items rated by many users. *Bandwagon* attack rates the I_f in a similar way with *random* attack, but the items in I_s will be rated with the maximum rating score. *Average over popular (AoP) x% attack* [9] only has I_s containing rated items randomly selected from the $x\%$ most popular item set. The chosen item is filled according to the normal distribution of the mean value and standard deviation of each rated item.

2.2. Shilling Attack Detection Methods

Shilling attack detection methods can be mainly categorized into two types: supervised and unsupervised detection models. The supervised detection usually uses Support Vector Machine (SVM) [10], C4.5 or k Nearest Neighbourhood (kNN) as a supervised classifier. The main idea of supervised detection is to design some discriminant features that could capture the difference between the malicious and genuine users. Based on the type of captured information, the existing features can be categorized as *profile similarity* [11], *rating behavior* [12] and *item distribution* [13] based features. Our previous research [5] indicated the difference between the behavior of selecting items to rate of genuine and malicious profiles. As calculating item correlation requires all information of the dataset, it is difficult for an adversary to obtain the accurate information on item correlation. The detection features based on the rated item correlation is a general and effective feature to increase the detection accuracy.

Although the supervised detection model has achieved a satisfying result, the labeled samples are required in the training process. Unsupervised detection model, which does not need any prior information, is less costly. K-mean, Probabilistic Latent Semantics Analysis (PLSA) [14] and Principle Component Analysis (PCA) [15], are applied to separate user profiles into clusters based on different discriminant characteristics in unsupervised detection models. For example, PCA is based on the observation that the malicious profiles are highly correlated with each other and have high similarity with a large number of genuine profiles.

3. Proposed Rated Item Correlation Measurement based Unsupervised Learning

As shown in our previous study [4], the rated item correlation measurement indicates the difference between real and malicious user profiles effectively. This section discusses how item correlation can be used in the unsupervised learning method. The measurement of the rated item correlation is firstly introduced in Section 3.1, and the proposed algorithm is then discussed in detail in Section 3.2.

3.1. Rated Item Correlation Measurement

The rated item correlation measurement, which quantifies the correlation between rated items in a user profile, was pro-

posed in [4]. In the given recommender system database, each row represents a user profile $u_i = [r_{i1}, r_{i2}..., r_{im}]$, where i represents the ith user profile, $i = 1, 2, ...n$, in which n is the number of users in the dataset, and m is the number of items considered in the system. The rating value of the item j given by the user i is defined as r_{ij}. If $r_{ij} = \emptyset$, the user i does not rate the item j.

U_i is defined as the set which only contains the items that have rated by user i, *i.e.,* $U_i = [s_j | r_{ij} \neq \emptyset, 1 \leq j \leq m]$. We defined $C_i^x = CA(x,i)$, in which $i = 1, 2, ...m$ and CA denotes the Cosine Association (CA) shown in (1). CA, which is an association rule used in data mining [16], indicates the possibility for a user to rate the item i and j at the same time without without considering their rating values. CA is chosen as it reduces the influence of the imbalance rating users.

$$CA(i,j) = \sqrt{\frac{|U_{i,j}|^2}{|U_i||U_j|}} \tag{1}$$

where $|U_{i,j}|$ represents the number of users who rate both item i and j, and $|U_i|$ denotes the number of users who rate i. CA is a symmetric function, *i.e.,* $CA(i,j) = CA(j,i)$. The range of CA is from 0 to 1.

Then all the calculated C_i^x are sorted in ascending orders, such as $C_{s_1}^x, C_{s_2}^x, ..., C_{s_m}^x$, where $C_{s_i}^x \leq C_{s_j}^x$ and $1 \leq i < j \leq m$. We only considered a particular interval of the distribution of the correlation, and the interval was defined as a percentage pair $(p_l\%, p_r\%)$, in which p_l and p_r is the parameter we will discuss in the experiment. The equation of the calculation of rated item correlation are shown as follow:

$$S_{(p_l,p_r)}^x = E_{\lfloor p_l \times m \rfloor \leq i \leq \lceil p_r \times m \rceil}(C_{s_i}^x). \tag{2}$$

Three features summarize the information in different arranges of the distribution. In [4], three arranges, including $(0\%, 20\%)$, $(40\%, 60\%)$, and $(80\%, 100\%)$, are determined by ad-hoc. In this paper, we will calculate the rated item correlation R for each profile as the discriminant information. The impact on the discriminant ability when changing p_l and p_r will be discuss in this paper. The best percentage pair $(p_l\%, p_r\%)$ will be selected according to the detection accuracy.

The equation of R are defined as follow:

$$R = E_{x \in U_i}(S_{(p_l\%, p_r\%)}^x) \tag{3}$$

where $p_l < p_r$, and $p_l, p_r \in [0, 100]$. The range of R is from 0 to 1. The larger value of r indicates those rated items in a user profile are highly correlated with each other, vice versa.

Algorithm 1 Unsupervised detection model

Input: **R**: a user profile database; $CA(i,j)$: the item correlation matrix based on **R**; U_i: a set containing the items rated by the ith user; p_l: the left margin; p_r: the right margin; k: the number of malicious user.

Output: M_{index}: the index of malicious users

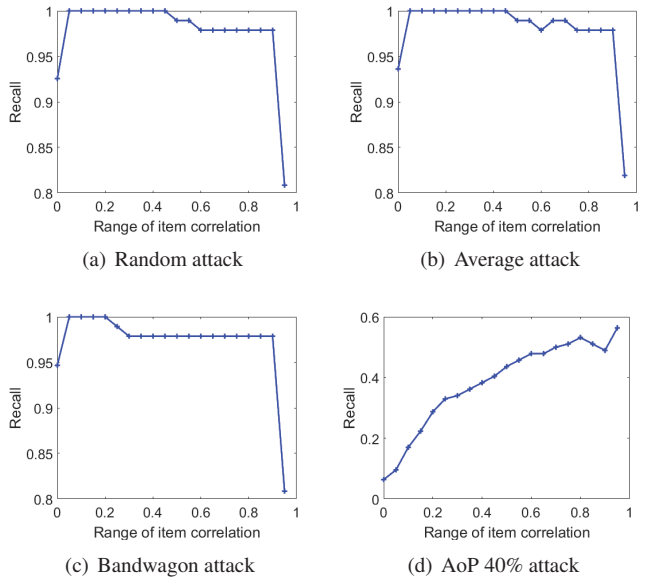
```
for each rated item x ∈ U_i do
    for each item i ∈ m do
        C_i^x ← CA(x, i);
    end for
    C_{s_i}^x ← Sort(C_i^x, asc);
    S_x ← E_{⌊p_l%×m⌋≤i≤⌈p_r%×m⌉}(C_{s_i}^x);
end for
R_x ← E(S_x) for x ∈ U_i;
INDEX_{R_x}, R_{s_x} ← Sort(R_x, asc);
M_index ← INDEX_{R_x}(1 : k)
Return
```

3.2. Proposed Algorithm

Genuine users tend to have their interest and rate the items with larger item correlation, while the rated item correlation is weak in malicious profiles [4]. With this prior knowledge, we proposed an unsupervised detection model which identifies the users with a low rated item correlation of their profiles as the malicious profiles.

Representing a given user profile database as **R**, our model is described in Algorithm 1. The inputs of the algorithm include the item correlation matrix which has been calculated according to equation 1 on each item pair in **R**, p_l and p_r determining the interval of the considered correlated item region. k is the threshold of the number of suspicious malicious user number. This algorithm firstly calculates the value of rated item correlation R for each input profile. Then the k users with the smallest R are considered as the malicious profiles by this algorithm.

The threshold of the number of suspicious malicious user number is a critical value which has a great impact on the detection accuracy. The parameter k of our model can be determined according to the first order difference of R_{s_x}, which is the sorted R of each input users. We defined the result of the first order difference vector of R_{s_x} as $Diff_R$. The previous research [4] showed that the distribution of the R of genuine profiles follows the normal distribution with a small variance. The first order difference calculated between the genuine users is in a small range of values. R of malicious profiles is usually much smaller than the one of genuine users. Therefore, a peak is expected to exist in $Diff_R$ if there exist the malicious profiles,



(a) Random attack (b) Average attack

(c) Bandwagon attack (d) AoP 40% attack

FIGURE 1. How recall change with different p_l

and the index of the peak is recognized as the number of suspicious malicious users k. The proper value of k is determined by $Diff_R$. If there exists a peak in the middle of $Diff_R$, the index of the peak is approximately recognized as the number malicious user. The detection accuracy according to this proposed unsupervised detection model will be measured in Section 4.3.

4. Experiment

In this section, we first introduce the settings of the experiment. Then we analyze the discriminant ability of the proposed unsupervised learning method with the rated item correlation in different ranges of correlated rated items, then the performance of the proposed unsupervised detection model is evaluated.

4.1. Experimental Setting

In the experiment, MovieLens 100K dataset[1] is used. The dataset is collected through the MovieLens website[2] in the seven-month period from September, 1997 to April, 1998, which contains 100,000 times of rating provided by 943 users. The dataset contains 1,682 movies in total, and the rating values are in the range of 1 to 5, in which a larger value represents a deeper degree of interest.

[1]https://grouplens.org/datasets/movielens/
[2]http://movielens.umn.edu

The shilling attack considered in our experiment are *random*, *average*, *bandwagon* and *AoP 40%* attacks mentioned in Section 2.1. Both of the *random* and *average* attack select all the items randomly from the whole dataset. As for bandwagon attack, 20% of the rated items are randomly selected from the popular item set, and the rest of the filler items are randomly selected from the whole dataset. *AoP 40%* attack selected the rated items from the 40% of the most popular item set.

4.2. Analysis of the Parameters in Rated Item Correlation Calculation

The aim of this section is to investigate how the parameters, p_l and p_r, affects the performance of our proposed model.

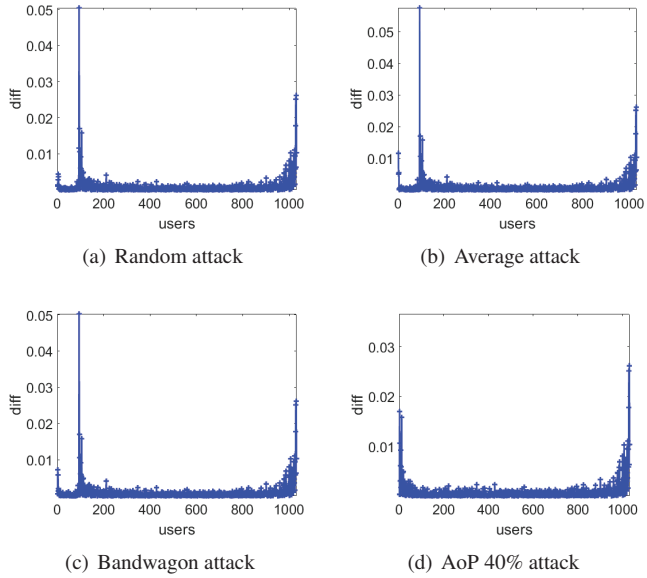
4.2.1. Experiment Setting

The attack size is set as 0.1, which means for each kind of attack, 94 malicious profiles are generated and injected into the dataset. The change interval of p_l is 5, and the interval length of $[p_l, p_r]$ is set as 5, which divides $[0, 100]$ into 20 regions. In this experiment, the parameter k is set as 94, which is the exact number of malicious profiles that injected into the system. The recall of the algorithm is measured with different inputs of p_l and p_r.

4.2.2. Results and Discussion

The results are shown in Figure 1. The x-axis represents the value of p_l while the y-axis represents the recall which is calculated according to the experimental setting. The results show that the recall changes with the rated item correlation with different intervals for each different kind of attack models.

Random and *average* attack use the same algorithm in selecting rated items. As the used item correlation measurement only relies on the chosen items but not the rated values, the rated item correlation of *random* and *average* attack is very similar, shown in Figures 1(a) and 1(b). When p_l is in the range of 5% to 90%, the recall is larger than 97.8% for both of *random* and *average* attack, *i.e.,* the detection is accurate. When p_l is in range of 5% to 45%, the recall reaches to 100%, which indicates the malicious profiles containing the randomly selected rated item obviously have smaller rated item correlation comparing with genuine users. However, when p_l is equal to 0% and 95%, the detection accuracy is lower than the other parameter. It indicates that when used this method, both of the smallest and largest of p_l is not suitable for sufficiently detect the malicious profiles. The result of bandwagon attack is shown in Figure 1(c), which have a similar characteristic to *random*



(a) Random attack (b) Average attack

(c) Bandwagon attack (d) AoP 40% attack

FIGURE 2. First order difference of the sorted vector of R

and *average* attack. When p_l is in the range of 5% to 20%, the detection accuracy reaches 100%. Therefore, a smaller p_l is suggested under *bandwagon* attack.

However, the detection accuracy is low under *AoP 40%* attack. When p_l is equal to 0.95, the recall reaches to its maximum value 57.4%. This is because when *AoP 40%* attack selects a large number of popular items in its profiles. The popular items are those items which were rated by numerous users. The item correlation between the popular items tends to be larger than the one of the other normal items. Therefore, our method is not capable to effectively detect *AoP 40%* attack.

4.3. Unsupervised Detection Model

The aim of this section is to evaluate the selection method of the threshold of the number of malicious users k and the detection accuracy under four different attack types.

4.3.1. Experiment Setting

We set $p_l = 10$ and $p_r = 30$ for *random* and *average* attack, $p_l = 5$ and $p_r = 25$ for *bandwagon* attack, and $p_l = 95$ and $p_r = 100$ for *AoP 40%* attack according to the previous subsection.

4.3.2. Result and Discussion

The first order difference $Diff_R$ of R_{s_x} is firstly calculated and shown in Figure 2 for each type of attack. The x-axis represents the value of $Diff_R$, while the y-axis represents the number of the input user profiles.

$Diff_R$ of *random* and *average* attack is similar to each other according to the same reason we discussed before. In Figure 2(a), the maximum value in the middle of the $Diff_R$ is 0.0489, and the index is 94, which exactly equal to the number of malicious users injected into the system. Similar results can be found in *average* attack and *bandwagon* attack, in which means the index of the middle peak for *average* attack is 94, and for *bandwagon* attack is 93. The middle peak does not exist in *AoP 40%*, which show the limitation of the proposed method. In this experiment, the detection recall of the unsupervised detection model for *random*, *average* and *bandwagon* attack is 100%, 100% and 98.9%, which suggests that the proposed detection model can achieve a great detection performance without any labeled information.

5. Conclusion

The efficient shilling attack detection method is the critical issue to ensure the robustness of recommender systems. Our previous research indicates a satisfying detection accuracy can be achieved by using the rated item correlation as the supervised detection feature. However, as the labeled samples sometimes are obtained difficulty, an unsupervised detection model is more preferable. With the prior knowledge that the rated item correlation of the malicious users is obviously smaller than the genuine users, an unsupervised detection model based on the rated item correlation is proposed. In this model, the users with the small value of the rated item correlation are suspected and treated as a malicious profile. The threshold of the number of the malicious users is determined through taking the first order difference of the rated item correlation of each user. We evaluate the detection performance using the MovieLens 100K dataset, and the experimental results suggest that our model achieves an accurate result in detecting *random*, *average* and *bandwagon* attack. One possible future work is to provide the proposed unsupervised detection model with more information to improve the detection performance.

Acknowledgements

This paper is supported by the Natural Science Foundation of Guangdong Province, China (No. 2018A030313203), and the Fundamental Research Funds for the Central Universities (No. 2018ZD32).

References

[1] Desrosiers, Christian, and G. Karypis, A Comprehensive Survey of Neighborhood-based Recommendation Methods. Recommender Systems Handbook, Springer, US, pp. 107-144, 2011.

[2] Gunes, Ihsan, et al. "Shilling attacks against recommender systems: a comprehensive survey." Artificial Intelligence Review, 42.4:767-799, 2014.

[3] Ortega F, Hernando A, Bobadilla J, Kang JH, "Recommending items to group of users using matrix factorization based collaborative filtering.", Inf Sci, 345(C):313C324, 2016.

[4] Chen K, Patrick P. K. Chan, Daniel S. Yeung, "Shilling Attack Detection using Rated Item Correlation for Collaborative Filtering", Proceeding of SMC2018 Conference, Miyazaki, in press, 2018.

[5] Chen K, Patrick P. K. Chan, et al. "Shilling Attack Based on Item Popularity and Rated Item Correlation against Collaborative Filtering." International Journal of Machine Learning and Cybernetics, pp. 1-13, 2018.

[6] Lam, Shyong K., and J. Riedl. "Shilling recommender systems for fun and profit." International Conference on World Wide Web ACM, pp. 393-402, 2004.

[7] P. Kaur and S. Goel, "Shilling attack models in recommender system," International Conference on Inventive Computation Technologies (ICICT), Coimbatore, pp. 1-5, 2016.

[8] Lam SK, Riedl J, "Shilling recommender systems for fun and profit." Int Conf World Wide Web, Vol 30, pp. 393C402, 2004.

[9] Seminario, Carlos E., and D. C. Wilson. "Attacking item-based recommender systems with power items." ACM Conference on Recommender Systems ACM, pp. 57-64, 2014.

[10] Zhou W, Wen J, Xiong Q, et al. "SVM-TIA a shilling attack detection method based on SVM and target item analysis in recommender systems." Neurocomputing, 210.C(2016):197-205, 2016.

[11] Chirita, Paul Alexandru, W. Nejdl, and C. Zamfir. Preventing shilling attacks in online recommender systems. pp. 67-74, 2005.

[12] Burke, Robin, et al. "Classification features for attack detection in collaborative recommender systems." ACM SIGKDD International Conference on Knowledge Discovery and Data Mining ACM, pp. 542-547, 2006.

[13] Zhang, Fuzhi, and Q. Zhou. "HHTCSVM: An online method for detecting profile injection attacks in collaborative recommender systems." Knowledge-Based Systems 65.4(2014):96-105, 2014.

[14] Hurley N, Cheng Z, Zhang M. "Statistical attack detection." ACM Conference on Recommender Systems, New York, pp. 149-156, October 2009.

[15] Mehta, Bhaskar, and W. Nejdl. "Unsupervised strategies for shilling detection and robust collaborative filtering." User Modeling and User-Adapted Interaction, 19.1-2(2009):65-97, 2009.

[16] Agrawal, Rakesh, T. Imielinski, and A. N. Swami. "Mining Association Rules Between Sets of Items in Large Databases." SIGMOD Conference. Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data. (1993):207–216, 1993.