Meta-Path and Matrix Factorization Based Shilling Detection for Collaborate Filtering

Xin Zhang[1,2], Hong Xiang[1,2(✉)], and Yuqi Song[1,2]

[1] Key Laboratory of Dependable Service Computing in Cyber Physical Society, Chongqing University, Ministry of Education, Chongqing, China
{zhang.x,xianghong,songyq}@cqu.edu.cn
[2] School of Big Data and Software Engineering, Chongqing University, Chongqing, China

Abstract. Nowadays, collaborative filtering methods have been widely applied to E-commerce platforms. However, due to its openness, a large number of spammers attack those systems to manipulate the recommendation results to earn huge profits. The shilling attack has become a major threat to collaborative filtering systems. Therefore, effectively detecting shilling attacks is a crucial task. Most existing detection methods based on statistical-based features or unsupervised methods rely on a priori knowledge about attack size. Besides, the majority of work focuses on rating attack and ignore the relation attack. In this paper, motivated by the success of heterogeneous information network and oriented towards the hybrid attack, we propose an approach DMD to detect shilling attack based on meta-path and matrix factorization. At first, we concatenate the user-item bipartite network and user-user relation network as a whole. Next, we design several meta-paths to guide the random walk to product node sequences and utilize the skip-gram model to generate user embeddings. Meanwhile, users' latent factors are decomposed by matrix factorization. Finally, we incorporate these embeddings and factors to joint train the detector. Extensive experimental analysis on two public datasets demonstrate the superiority of the proposed method and show the effectiveness of different attack strategies and various attack sizes.

Keywords: Shilling detection · Meta-path · Hybrid attack · Heterogeneous information network · Collaborative filtering

1 Introduction

In recent years, with the proliferation of the Internet, a large number of E-commerce platforms are advancing by leaps and bounds, such as Amazon and Taobao. However, due to the wide range of products, it is difficult for users to find what they are truly interested in. Therefore, those platforms use recommender

H. Gao et al. (Eds.): CollaborateCom 2018, LNICST 268, pp. 3–16, 2019.
https://doi.org/10.1007/978-3-030-12981-1_1

system to provide potential personalized products for their customers to alleviate the above information overload problem. And the most prevalent recommended method is collaborative filtering, which recommends items based on purchase behavior of target customer and other customers with similar preference.

Nonetheless, due to the openness of collaborative filtering recommender systems, numerous malicious users (named spammers) can inject biased profiles (namely shilling profiles) into systems to manipulated the recommendation results for authentic users for gaining more profit. Meanwhile, according to various aspirations, some merchants resort to improving the recommendation of their products via increasing the ratings of their own products while another seller endeavor to decrease scores of competitive commodities, and the former called push attack and the latter called nuke attack. Such fraudulent actions are shilling attacks which badly change the recommendation results and affect the decision of the prospective consumers. In consequence, how to detect shilling attacks is a core task of improving the robustness of recommender systems.

Generally, the shilling detection can be regarded as a binary classification problem, which means we need to identify a user is a malicious user (named spammer) or authentic user through his/her profiles. To this end, the main point of this problem is to analyze and model the characteristic of users. Up to now, although dozens of notable works have been down to detect shilling attacks, most of them highly rely on the statistical manners, which may fail in revealing the fine-grained interactions between users and items. Besides, as the collaborative filtering relies on users preference, the relations between users also can make effort to recommendation and attack, but there is little work pay attention to it.

According to above intuition, to dig the interactions between users and items and explore the relations among users for detection, in this paper, we propose a shilling detection algorithm named DMD (Double M Detector). We use the matrix factorization to decompose the user-item rating matrix to obtain the latent factors, while design several meaningful meta-paths based on Heterogeneous Information Network (HIN) according to network characteristics, such as degree, hindex and coreness, to represent users' embeddings of latent relations. Furthermore, joint training the detector via above latent factors to predict the label of users.

The main contributions of this paper are summarized as follows:

- We propose a novel method DMD which exploits the interactions among user-item and user-user based on HIN to detect shilling attacks for collaborative filtering recommender systems;
- We not only focus to detect the rating attacks, but also pay attention to relation attacks, and the proposed DMD is effective for hybrid attacks.
- With extensive experiments on the real-world Amazon dataset and simulated FilmTrust dataset, we evaluate and compare the performance of the method with other methods to show its effectiveness.

The remainder of this paper is organized as follows. Section 2 reviews the related work of shilling detection. Section 3 presents the preliminaries about shilling attack models and the proposed method. The illustration of DMD in

detail is shown in Sect. 4. In Sect. 5, we conduct experiments on two datasets. Finally, Sect. 6 concludes the whole paper.

2 Related Work

In collaborative filtering recommender system, the key vulnerabilities derive from the openness of itself and the high reliance on user profiles. To alleviate the shilling attack and reinforce the robustness of collaborative filtering recommender system, many researchers engaged in the field of shilling detection. According to the intent of spammers, to promote items or prevent items from being recommended, attacks can be categorized into two types: the push attack and the nuke attack. As the basic principle of the two kinds of attacks is the same, the most research pays more attention to the push attack.

In the early stage, researchers mainly focused on statistical analysis methods to detect anomalies caused by suspicious ratings. For example, there was some work relied on item average ratings [1] or leveraged Neyman-Person statistical detection theory [2] and so forth. Meanwhile, a lot of research has been undertaken to employ supervised learning for detection, those classifiers are trained through labels information. For instance, a detector based on the average similarity and the Rating Deviation from Mean Agreement (RDMA) metric was presented in [3], a decision-tree based proposed in [4]. More specifically, Williams et al. [5] proposed several generic and attack type-specific attributes, and trained three supervised machine learning algorithms to detect shilling attacks. Recently, Li et al. [6] developed an algorithm, which explored item's popularity degree features; Zhou et al. [7] first used an SVM-based classifier to obtain an amount of suspicious profiles, secondly, removed the genuine profiles from the set via target item analysis method, and Dou et al. [8] proposed a CoDetector model, which jointly decomposes the user-item matrix and the user-user co-occurrence matrix.

Although supervised methods usually could train a good performance detector, it totally depends on labeled samples which increase the number of experts and time consuming to a large extent. Therefore, unsupervised models are utilized in the shilling detection, which are more applicable to real scenarios. The early classical approach is PCASelectUsers [9], which exploited the principal component analysis on the rating records. Lately, Zhang et al. [10] presented a unified framework based on the idea of label propagation, but it requires to set the number of spammers as the initial seed users.

Apart from the above methods, some semi-supervised models have also been explored in shilling detection. A hybrid shilling attack detector was proposed by Wu [11], which collects many detection metrics for selecting features via a wrapper called MC-Relief and the semi-supervised Naive Bayes for classification. In [12], a model based on PU-Learning which relies on a few positive labels and much unlabeled to construct a classifier iteratively was introduced.

The above-mentioned methods all have some limitations: supervised and semi-supervised detection methods are restricted by labeled samples, unsupervised detection methods need some prior knowledge about attacks to guarantee

their performance. In addition, some methods are only suitable for detecting known types of attacks, when handling some new or unknown attacks, the performance is poor. Furthermore, most of them focused on rating information rather than relations between users.

3 Preliminaries

3.1 Shilling Attack Models

According to information that attackers used [13], we classify the shilling attack into three broad categories: rating attack, relation attack and hybrid attack. The definitions of those attacks are shown in Table 1.

Table 1. The definition of three types of attack.

Types	Definition
Rating attack	Injecting biased rating profiles to manipulate the recommendation results
Relation attack	Through link farming to influence user's social relationship and distort neighbors' preferences
Hybrid attack	Fusing ratings and relationships to enlarge destructiveness in recommender systems

In the three types of attack, rating attack is a typical and most common forms to affect the recommendation, and despite the fact that relation attack usually aims at social network rather than recommender system, but it can be used as an auxiliary to enhance destructiveness. In consequence, in this paper, we intend to integrate the rating information and users' relationship to detect the shilling attack.

In order to attack the target product and in the same time behave like an authentic user to avoid being detected, spammers always use attack model and generate attack profiles based on knowledge of recommender system. The general rating profile can be divided into four parts. Meanwhile, we combine the relation profile with two segments into the rating profile to form the hybrid profile, which is depicted in Fig. 1. Specifically, the explanation of the above six parts are listed below.

- **Target item** (I^T) indicates the items that spammers design to recommend more often.
- **Selected items** (I^S) are those spammers used to make the relationship with authentic users.
- **Filler items** (I^F) are some items for spammers to disguise themselves as authentic users.

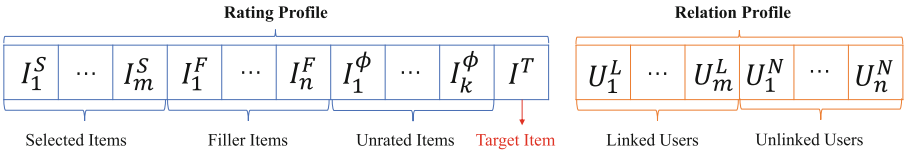

Fig. 1. The general framework of hybrid profile.

- **Unrelated items** (I^{Φ}) stand for items that spammers do not rate, which forms the majority of rating profile.
- **Linked users** (U^{L}) are those users that spammers try to establish a relationship with.
- **Unlinked users** (U^{N}) imply that there is no direct social link between them and spammers, which account for the largest part in relation profile.

In accordance with different attack strategies, rating attack models are categorized into four types, namely random attack, average attack, bandwagon attack and segment attack. Similarly, relation attack models are classified into two categories: random link attack and targeted link attack. Hence, by bridging rating and relation attacks, the hybrid attacks are composed of eight kinds of model: R-random attack, R-average attack, R-bandwagon attack, R-segment attack, T-random attack, T-average attack, T-bandwagon attack and T-segment attack. Table 2 describes these attack models.

Table 2. The features of the attack models.

Models	I^S	I^F	I^T
Random attack	$\emptyset$	randomly chosen items, $r(I_i^F) = r_{random}$	*push*: $r(I^T) = r_{max}$ *nuke*: $r(I^T) = r_{min}$
Average attack	$\emptyset$	randomly chosen items, $r(I_i^F) = r_{mean}$	
Bandwagon attack	popular items, $r(I_i^S) = r_{max}$	randomly chosen items, $r(I_i^F) = r_{random}$	
Segment attack	like the target item, $r(I_i^S) = r_{max}$	randomly chosen items, $r(I_i^F) = r_{min}$	
Random link attack	U^L: randomly chosen users		
Targeted link attack	U^L: users chosen according to the specific attack plan		

From above table we can make a summary that the target items are always rated the highest rating, the filler items are randomly chosen and rated with different strategies, sometimes, the selected items are not required. As for users, the linked ones are usually chosen randomly but targeted link may according to the specific plan.

3.2 Heterogeneous Information Network

To detect the hybrid attack and motivated the existing studied [14–16], we consider to concatenate the user-item bipartite network and user-user social network as a whole to a Heterogeneous Information Network [17].

Definition 1. **Heterogeneous Information Network**: A graph $H = (V, E, T)$ in which each node v and each link e is tied via their mapping function $\phi(v)V \rightarrow T_V$ and $\phi(e)E \rightarrow T_E$, respectively. Meanwhile, T_V and T_E refer to the types of objects and relations in V and E, and $|T_V| + |T_E| > 2$. This graph H is named HIN. Figure 2 is an illustration of our proposed HIN, where three types of nodes and two types of edge are involved.

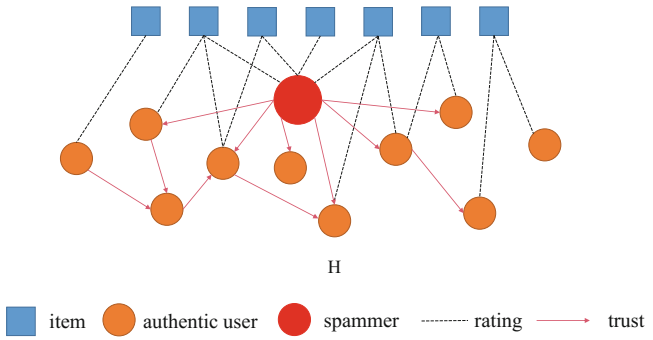

Fig. 2. The heterogeneous information network H.

3.3 Meta-Path

Inspired by the success of network embedding models [15,18], we will design some meta-paths over the HIN to capture the potential characteristics behind spammers.

Definition 2. **Meta-path** [19]: A meta-path scheme P is defined as a path that is denoted in the form of $V_1 \xrightarrow{R_1} V_2 \xrightarrow{R_2} \cdots \xrightarrow{R_{l-1}} V_l$, where $R = R_1 \circ R_2 \cdots \circ R_l$ defines the composite relations from its first type V_1 to the last type V_l.

4 The Proposed Method

In this paper, we propose a meta-path and matrix factorization based method (DMD) to spot shilling attack, and the detection framework of DMD is depicted in Fig. 3.

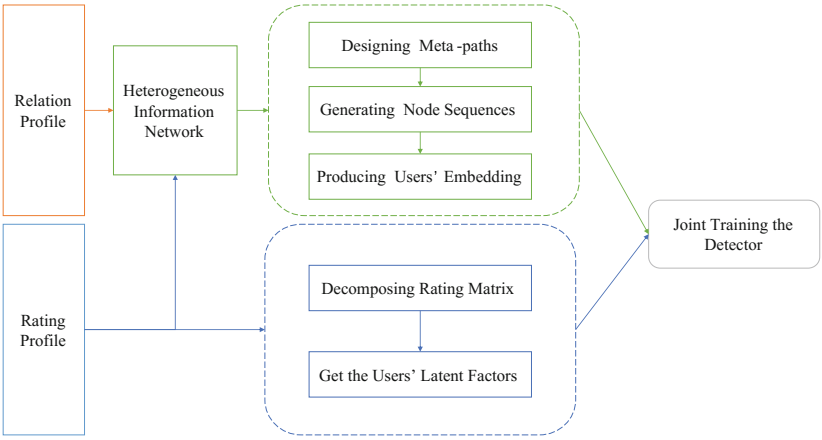

Fig. 3. The framework of DMD

4.1 Exploring Meta-Paths to Get Users' Embedding over HIN

In order to detect the hybrid attack, the user-item bipartite network and user-user social network are cultivated as a whole HIN. For mining anomalous behavior pattern more precisely, we design four meta-paths to model the relations among users according to three network features, as shown in Table 3.

Table 3. The designed meta-paths.

Path	Description
User → Item → User (UIU)	Explore users who rated the same items
User → Degree → User (UDU)	Linked users who have the same degree
User → H-index → User (UHU)	Linked users who have the same Hindex
User → Coreness → User (UCU)	Linked users who have the same coreness

These meta-paths can be used to find a pair of entities that similar but are distant from each other on the original bipartite network and the social network. Furthermore, three network features are used to link users. **Degree** which counts the number of current user linked neighbors, it is the simplest way to measure the importance of a node. **H-index** which was originally used to measure the citation impact of a scholar or a journal [20] and it is defined as the maximum value h such that there exists at least h papers, each with citation count $\geq h$. Here, the H-index of a node is defined to be the maximum value h such that there exists at least h neighbors of degree no less than h. **Coreness** [21] is measured by k-core decomposition [22], and a larger coreness value indicates that the node is more centrally located in the network.

Next, those meta-paths are utilized to conduct random walks to generate a number of node sequences. As most social relations are noisy, we use biased probability to create node sequences. Given a meta-path schema $\mathcal{P} = V_1 \xrightarrow{R_1} V_2 \xrightarrow{R_2} \cdots \xrightarrow{R_{l-1}} V_l$, the transition probability at step i is as follows:

$$p(v^{i+1}|v_t^i, \mathcal{P}) = \begin{cases} \frac{1}{|N_{t+1}(v_t^i)|} & (v^{i+1}, v_t^i) \in rated \\ \frac{\psi(v^{i+1}, v_t^i)}{\sum_{v' \in N_{t+1}(v_t^i)} \psi(v', v_t^i)} & (v^{i+1}, v_t^i) \in trust \\ 0 & (v^{i+1}, v_t^i) \notin E \end{cases} \tag{1}$$

where $v_t^i \in V_t$, $N_{t+1}(v_i^t)$ indicates the V_{t+1} kind of neighborhood of node v_t^i, and

$$\psi(v^{i+1}, v_t^i) = |N_{t+1}(v^{i+1}) \cap N_{t+1}(v_t^i)|. \tag{2}$$

According to the definition, at each step of the random walk, the successor node type is decided by the pre-defined meta-path $\mathcal{P}$ at each step in a random walk. When $V_t = U$ and $V_{t+1} = I$ (or the inverse), the successor node is chosen. However, if $V_t = V_{t+1} = U$ (or the inverse), the successor node is selected by the amount of overlapped neighbors with the current node.

In the next stage, as the collected random walks consist of different types of nodes, we feed it to the heterogeneous Skip-Gram model proposed by [15], for learning node embeddings $X \in \mathbb{R}^{\|V\| \times d}$. Formally, given a meta-path guided node sequence and the current node v^i, the objective function is:

$$\arg\max_{\theta} \sum_{v \in V} \sum_{v_t^n \in C(v^i)} \log p(v_t^n | v^i; \theta), \tag{3}$$

where $C(v^i)$ is the context information of v^i with the window size w and $p(v_t^n|v^i;\theta)$ is defined as a heterogeneous softmax function:

$$p(v_t^n | v^i; \theta) = \frac{e^{x_{v_t^n} \cdot x_{v^i}}}{\sum_{v \in V_t} e^{x_v \cdot x_{v^i}}}, \tag{4}$$

in which x_v is the v^{th} row of X, representing the embedding vector of node v, and V_t is the node set of type t in H.

However, calculating Eq. 4 is still computationally expensive, to accelerate the optimization, we adopt negative sampling [23] for the learning task. Given the type of the node in $C(v^i)$ and the negative sample size M, we randomly select M nodes with the same type label from V for the construction of softmax and then update Eq. 4 by the following objective:

$$\mathcal{O}(X) = \log \sigma(x_{v_t^n} \cdot x_{v^i}) + \sum_{m=1}^{M} \mathbb{E}_{v_t^m \sim P_t(v_t)} [\log \sigma(-x_{v_t^m} \cdot x_{v^i})], \tag{5}$$

where $\sigma(x) = \frac{1}{1+e^{-x}}$ and the sampling distribution $P_t(v_t)$ specified by the node type of v_t^n is a uniform distribution.

4.2 Decomposing Matrix for Users' Latent Factors

Matrix factorization (MF) is a basic method in collaborative filtering which uncovers the latent features underlying the interactions between users and items by mapping both users and items into a low-dimensional latent-factor space [24]. To capture the implicit features in rating profile, we use MF to gain users' latent factors. The objective function of this step is:

$$L = \sum_{u,i} (y_{ui} - p_u^T q_i)^2 + \lambda(\sum_u \|p_u\|^2 + \sum_i \|q_i\|^2), \tag{6}$$

where p_u and q_i indicate user and item latent factors respectively, y_{ui} means the rating record created by user u on item i, and $p_u^T q_i$ is a predictive value. The parameter λ denotes the magnitudes of the latent factors to prevent overfitting.

4.3 Joint Training Based on Above Embeddings

When the node representations are obtained by performing a stochastic gradient ascent method on Eq. 5 and user latent factors are decomposed by Eq. 6, we incorporate them into a random forest model [25], which is an ensemble learning method by constructing a multitude of decision trees at training time and outputting the class that is the mode of the class of the individual trees. After the training, we can obtain the detector to identify spammers from unlabeled users.

Algorithms 1 shows the overall process of our proposed method DMD.

Algorithm 1. DMD

Input: User Label U, user-time rating matrix R, The heterogeneous network $H = (V, E, T)$ combined by user-item bipartite graph and user social graph, a meta-path schema $\mathcal{P}$, #walks per user n, walk length l, embedding dimension X, window size w, #negative samples M

Output: Labels of users to be recognized

```
initialize node embeddings X
for user i in V do
    for j = 1 → n do
        MP = MetaPathRandomWalk(H, P, i, l)
        X = HeterogeneousSkipGram(X,MP,w)
while notConverged do
    decompose R
    update user latent vectors P and item latent vectors Q
joint embeddings X and user latent vectors P training the detector DMD based on U
use DMD to predict user labels
```

5 Experiments

In this section, we present the experimental work. Firstly, two datasets and three metrics will be introduced. Next, we conduct experiments to evaluate the effectiveness of our detector and compare it with other detection methods. Furthermore, several experiments will be done to verify whether the DMD can handle different attacks.

5.1 Datasets and Metrics

We adopt two real-world datasets in experiments: Amazon dataset [26] which includes spammers per se and we extracted user social relationships from candidate groups; FilmTrust [27] is a typical dataset for recommendation without spammers, therefore, we inject spammers based on attack models for detecting. The detailed statistics of those datasets are shown in Table 4. To tune the methods included, we use 80% of the data as the training set and the others as the test set, meanwhile, we randomly select 10% from training set as the validation set.

Table 4. The datasets

Dataset	#Users	#Items	#Ratings	#Relations	# Spammers
Amazon	4,902	21,394	60,000	78,418	1,937
FilmTrust	1,508	2,071	35,479	1,853	0

The experiments were conducted by 5-fold cross validation 10 times, where average values of each set of trials were generated to represent the final results. We adopt the three frequently-used evaluation metrics, i.e., *Precision*, *Recall* and *F-measure* for performance evaluation.

$$Precision = \frac{TP}{(TP + FP)} \tag{7}$$

$$Recall = \frac{TP}{(TP + FN)} \tag{8}$$

$$F - measure = \frac{2 \times Precision \times Recall}{(Precision + Recall)} \tag{9}$$

where P and N represent positive samples and negative samples. The true positive (TP) sample means predicted and actual labels both are positive. If the predicted label is positive, and the actual label is negative, the instance is a false positive (FP) sample. Likewise, false negative (FN) means that the predicted label is negative, but the actual label is positive.

5.2 Experimental Results

Detection Performance. The performance of DMD is compared with DegreeSAD [6], FAP [10], SemiSAD [28] and CoDetector [8]. Among them, DegreeSAD is a supervised method based on popularity degree features, FAP is an unsupervised method via tag probabilistic propagation, FAP is a semi-supervised method. CoDetector, as a supervised method bridging factorization and user embedding, which is the most similar approach to DMD, but it did not explore the social relations. In addition, we inject R-random hybrid attack to the FilmTrust dataset and the rating attack size, rating attack filler size and relation attack size are set to 10%, 5% and 0.2% respectively. The experimental result is shown in Table 5.

Table 5. Performance comparison of our methods and other methods.

	Metric	DegreeSAD	FAP	SemiSAD	CoDetector	DMD
Amazon	Precision	0.7145	0.8931	0.6037	0.8812	**0.9336**
	Recall	0.6184	0.7290	0.6203	0.8915	**0.9084**
	F-measure	0.6626	0.8028	0.6138	0.8863	**0.9208**
FilmTrust	Precision	0.8125	0.8367	0.8333	0.7600	**0.921**
	Recall	0.9286	0.8662	0.7407	0.8636	**0.9347**
	F-measure	0.8667	0.8512	0.7843	0.8085	**0.9269**

We can make the following observations from the table: in all cases, our proposed model DMD outperform all the compared baseline methods. Specifically, on Amazon, the precision, recall and f-measure all reach 90%, and the improvements on them are 4.52%, 1.90% and 3.89%, respectively. On the FilmTrust dataset which injected hybrid attack with random rating and random link, the three metrics all reach highest values, and the precision increases 10.08% and the f-measure increases 6.95%. Therefore, the DMD not only can detect rating attack but also have the ability to handle the relation attack and hybrid attack. In summary, the performance of DMD has a significant advantage over the other four methods, and it shows the effectiveness and robustness of DMD whether in the simulated dataset or real world dataset.

Detection of Simulated Attack. To further demonstrate that our proposed method has good performance to cope with different attack strategies and various attack sizes, we especially attack the FilmTrust dataset manually according to the definition of attack models. As for relation attack, we inject the random link profile with 0.1%, 0.2% and 0.5% link size. For rating attack, the random attack, average attack and bandwagon attack are injected with 5% and 10% attack size. It should be noted that the original users are labeled as normal users and injected ones are spammers. After that, we use DMD to detect those simulated spammers. The results of the experiment are shown in Table 6.

Table 6. Detection results of hybrid attacks on FilmTrust

Link size	Metric	Random		Average		Bandwagon	
		5%	10%	5%	10%	5%	10%
0.1%	Precision	0.9494	0.9081	0.9381	0.933	0.8584	0.8932
	Recall	0.9019	0.8929	0.8903	0.9582	0.8955	0.9007
	F-measure	0.9219	0.8983	0.9068	0.9446	0.8766	0.8969
0.2%	Precision	0.9433	0.921	0.9382	0.9531	0.8876	0.9088
	Recall	0.8259	0.9347	**0.9767**	0.9738	0.9039	0.9263
	F-measure	0.8636	0.9269	0.9553	0.9624	0.8957	0.9167
0.5%	Precision	0.9214	0.8897	0.9192	**0.9724**	0.9269	0.9085
	Recall	0.9167	0.9649	0.8789	0.9757	0.8717	0.9221
	F-measure	0.9162	0.9255	0.8958	**0.9728**	0.8979	0.9144

As shown in Table 6, facing these hybrid attacks, the majority of detection results more than 0.9 and the highest precision, recall and f-measure reach 0.9724, 0.9767 and 0.9728, respectively. As for the different rating attack strategies, the MDM achieve the best performance in average attack detection.

6 Conclusion

In this paper, we proposed a novel shilling detection method DMD based on the meta-path and matrix factorization for collaborative filtering recommender system. Firstly, we incorporate the user-item rating network and user-user relation network as a whole heterogeneous information network and design four meta-paths to capture the undirectly links between users. Afterward, node sequences are produced guided by random walk according to above-mentioned meta-paths. Next, we use the skip-gram model to generate user embedding. In the meantime, we decompose the rating matrix based on matrix factorization to gain the users' latent factors. Finally, using embedding and factors are used to joint train the detector. Experimental results on one real-world public dataset and a simulated dataset show the DMD improve the preference of detecting spammers. In addition, it is not only effective for the rating attack but also has good ability to detect the hybrid attack.

Acknowledgments. The work is supported by the Fundamental Research Funds for the Central Universities (106112017CDJXSYY0002).

References

1. Bhaumik, R., Williams, C., Mobasher, B., Burke, R.: Securing collaborative filtering against malicious attacks through anomaly detection. In: Proceedings of the 4th Workshop on Intelligent Techniques for Web Personalization (ITWP 2006), Boston, vol. 6, p. 10 (2006)
2. Hurley, N., Cheng, Z., Zhang, M.: Statistical attack detection. In: Proceedings of the third ACM Conference on Recommender Systems, pp. 149–156. ACM (2009)
3. Chirita, P.-A., Nejdl, W., Zamfir, C.: 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. ACM (2005)
4. Williams, C., Mobasher, B.: Profile injection attack detection for securing collaborative recommender systems. DePaul University CTI Technical Report, pp. 1–47 (2006)
5. Williams, C.A., Mobasher, B., Burke, R.: Defending recommender systems: detection of profile injection attacks. Serv. Oriented Comput. Appl. **1**(3), 157–170 (2007)
6. Li, W., Gao, M., Li, H., Xiong, Q., Wen, J., Ling, B.: A shilling attack detection algorithm based on popularity degree features. Acta Autom. Sinica **41**(9), 1563–1576 (2015)
7. Zhou, W., Wen, J., Xiong, Q., Gao, M., Zeng, J.: SVM-TIA a shilling attack detection method based on svm and target item analysis in recommender systems. Neurocomputing **210**, 197–205 (2016)
8. Dou, T., Yu, J., Xiong, Q., Gao, M., Song, Y., Fang, Q.: Collaborative shilling detection bridging factorization and user embedding. In: Romdhani, I., Shu, L., Takahiro, H., Zhou, Z., Gordon, T., Zeng, D. (eds.) CollaborateCom 2017. LNICST, vol. 252, pp. 459–469. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00916-8_43
9. Mehta, B., Nejdl, W.: Unsupervised strategies for shilling detection and robust collaborative filtering. User Model. User-Adap. Interact. **19**(1–2), 65–97 (2009)
10. Zhang, Y., Tan, Y., Zhang, M., Liu, Y., Chua, T.-S., Ma, S.: Catch the black sheep: Unified framework for shilling attack detection based on fraudulent action propagation. In: IJCAI, pp. 2408–2414 (2015)
11. Wu, Z., Wu, J., Cao, J., Tao, D.: HySAD: a semi-supervised hybrid shilling attack detector for trustworthy product recommendation. In: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 985–993. ACM (2012)
12. Wu, Z., Wang, Y., Wang, Y., Wu, J., Cao, J., Zhang, L.: Spammers detection from product reviews: a hybrid model. In: 2015 IEEE International Conference on Data Mining (ICDM), pp. 1039–1044. IEEE (2015)
13. Junliang, Y., Gao, M., Rong, W., Li, W., Xiong, Q., Wen, J.: Hybrid attacks on model-based social recommender systems. Phys. A: Stati. Mech. Appl. **483**, 171–181 (2017)
14. Yuan, Q., Chen, L., Zhao, S.: Factorization vs. regularization: fusing heterogeneous social relationships in top-n recommendation. In: Proceedings of the fifth ACM Conference on Recommender Systems, pp. 245–252. ACM (2011)
15. Dong, Y., Chawla, N.V., Swami, A.: metapath2vec: scalable representation learning for heterogeneous networks. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 135–144. ACM (2017)

16. Song, Y., Gao, M., Yu, J., Xiong, Q.: Social recommendation based on implicit friends discovering via meta-path. In: Proceedings of the 30th International Conference on Tools with Artifical Intelligence (2018)
17. Sun, Y., Han, J.: Mining heterogeneous information networks: principles and methodologies. Synth. Lect. Data Min. Knowl. Discov. **3**(2), 1–159 (2012)
18. Perozzi, B., Al-Rfou, R., Skiena, S.: DeepWalk: online learning of social representations. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 701–710. ACM (2014)
19. Sun, Y., Han, J., Yan, X., Yu, P.S., Wu, T.: PathSim: meta path-based top-k similarity search in heterogeneous information networks. Proc. VLDB Endow. **4**(11), 992–1003 (2011)
20. Hirsch, J.E.: An index to quantify an individual's scientific research output. Proc. Nat. Acad. Sci. **102**(46), 16569–16572 (2005)
21. Lü, L., Zhou, T., Zhang, Q.-M., Stanley, H.E.: The h-index of a network node and its relation to degree and coreness. Nat. commun. **7**, 10168 (2016)
22. Dorogovtsev, S.N., Goltsev, A.V., Mendes, J.F.F.: K-core organization of complex networks. Phys. Rev. Lett. **96**(4), 040601 (2006)
23. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)
24. Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer **8**, 30–37 (2009)
25. Liaw, A., Wiener, M., et al.: Classification and regression by randomforest. R news **2**(3), 18–22 (2002)
26. Xu, C., Zhang, J., Chang, K., Long, C.: Uncovering collusive spammers in Chinese review websites. In: Proceedings of the 22nd ACM International Conference on Conference on Information and Knowledge Management, pp. 979–988. ACM (2013)
27. Guo, G., Zhang, J., Yorke-Smith, N.: A novel Bayesian similarity measure for recommender systems. In: IJCAI, pp. 2619–2625 (2013)
28. Cao, J., Wu, Z., Mao, B., Zhang, Y.: Shilling attack detection utilizing semi-supervised learning method for collaborative recommender system. World Wide Web **16**(5–6), 729–748 (2013)