Semantic Shilling Attack against Heterogeneous Information Network Based Recommend Systems

Yizhi REN[†], Zelong LI[†], Lifeng YUAN^{†a)}, Zhen ZHANG^{†b)}, Nonmembers, Chunhua SU^{††}, Member, Yujuan WANG[†], and Guohua WU[†], Nonmembers

SUMMARY The recommend system has been widely used in many web application areas such as e-commerce services. With the development of the recommend system, the HIN modeling method replaces the traditional bipartite graph modeling method to represent the recommend system. But several studies have already showed that recommend system is vulnerable to shilling attack (injecting attack). However, the effectiveness of how traditional shilling attack has rarely been studied directly in the HIN model. Moreover, no study has focused on how to enhance shilling attacks against HIN recommend system by using the high-level semantic information. This work analyzes the relationship between the high-level semantic information and the attacking effects in HIN recommend system. This work proves that attack results are proportional to the high-level semantic information. Therefore, we propose a heuristic attack method based on high-level semantic information, named Semantic Shilling Attack (SSA) on a HIN recommend system (HERec). This method injects a specific score into each selected item related to the target in semantics. It ensures transmitting the misleading information towards target items and normal users, and attempts to interfere with the effect of the recommend system. The experiment is dependent on two real-world datasets, and proves that the attacking effect is positively correlate with the number of meta-paths. The result shows that our method is more effective when compared with existing baseline algorithms.

key words: shilling attack, heterogeneous information network, recommend system, semantic information

1. Introduction

The recommend system (RS) enables users locating their needs from large resources. It greatly improves the users' experience, which is thereby regarded as a significant part of the online services. The traditional recommend method is based on users' preferences, items' characteristics, and other additional information (such as historical interaction information, etc.) to build the recommend model. The existing approaches include content-based methods, collaborative filtering methods, and hybrid methods [1]. With the development of deep learning, various researchers abstracted the non-linear relationships of the coding to obtain the deeper representation embeddings of the user/item [2]. Besides, several studies captured the relationships from the self-data (such as contextual information) to improve the

accuracy of the recommend system [2].

However, the traditional methods that model RS as a bipartite graph cannot solve the cold start problem caused by sparse data. Thus, some works used heterogeneous information network (HIN) to model RS and solved this problem. Shi et al. [3] modeled the recommend system as a weighted heterogeneous information network and proposed a personalized recommend method SemRec. Wang et al. [4] proposed the HueRec integrating network embedding. While, Fan et al. [5] proposed the MEIRec method with deep learning.

Moreover, as an open information system, it is obvious that the RS is vulnerable to shilling attack (injecting attack) [6]. In that case, attackers are able to reduce the effectiveness of the system by injecting fake users and ratings. Literature [7] proposed the random attack (RA), which gives the highest score to the target whereas fabricates random rating behaviors to other items. After that, O'Mahony et al. [8] proposed a push attack method associating with average scores, which help merchants increasing the exposure rate of their merchandises. Recent studies [9]–[12] proposed shilling attacks against RS based deep learning. Li et al. [9] focused on the perspective of matrix factorization. Fang et al. [10] proposed a shilling attack on the GCN recommend system. Deldjoo et al. [11] studied the impact of user activities from injecting attack, then improved the effect of injection. Chen et al. [12] proposed a shilling attack combining the similarity of products.

However, existing works mainly focus on the bipartite graph recommend system. And the effectiveness of the traditional shilling attack on the HIN recommend system has not been proven. In theory, compared with the bipartite graph RS, the HIN recommend system has more information available in the attack due to it converges more high-level semantics. Under this assumption, how to use the characteristics of high-level semantic propagation to design an attack method that meets the HIN recommend scenario is considered as a new problem. The challenges of this problem are the structure of HIN is more complicated than bipartite graph's, and the expansion of the attackers' chose space makes the process of determining filling items more difficult. In this work, We build several attack experiments on HERec [13] which is a HIN recommend method based on meta-paths [14]. Compared with bipartite-graph models, the high-level semantics can enhance the attacking effect in HIN. Our experiments show that the more meta-paths add,

Manuscript received May 21, 2021.

Manuscript revised July 14, 2021.

Manuscript publicized November 30, 2021.

[†]The authors are with School of Cyber Space Security, Hangzhou Dianzi University, Hangzhou, 310018 China.

^{††}The author is with the Division of Computer Science, University of Aizu, Aizuwakamatsu-shi, 965–8580 Japan.

a) E-mail: yuanlifeng@hdu.edu.cn (Corresponding author)

b) E-mail: zhangzhen@hdu.edu.cn (Corresponding author)

DOI: 10.1587/transinf.2021BCP0015

the better attack results we have. Meanwhile, we design a new shilling attack combining with semantics to determine the user's scoring of the target product. Attackers directly score to the target item and establish a connection with the selected semantic-related products under meta-paths, which affect the users' scoring of the target product. We verify the effectiveness of the method based on two real datasets, also discuss the impact on various attack sizes and targets. The results show that our attack method overcomes several traditional attack methods by improving the total score of the target.

The main contributions of this work are as follows:

- We analyze the characteristics of the HIN recommend system which uses and disseminates the high-level semantic information under attack scenario. As we know, this is the first work on this issue. We gradually add the meta-path containing different high-level semantics to the HIN recommend system on two real-world networks. Then we analyze the relationship between the high-level semantics and shilling attack effect. A further novel finding is that attacking effect is proportional to the number of meta-paths with different depths.
- We propose a new attack method using high-level semantic information to against the meta-paths-represented recommend algorithm. Compared with the existing methods, this is the first shilling attack that combines semantic information. The experiment proves that our method is better than the baseline method. We improve the item score by directing partly filling items to the same store or type of goods. Meanwhile, the cost of the attacker is reduced to a certain extent.

The rest of this paper is organized as follows. In Sect. 2, we summarize the existing researches from related areas. Section 3 reviews the main components of the victim model and presents some preliminary knowledge. Then, we define the attack model and introduce the Semantic Shilling Attack against HIN recommend system in Sect. 4. In Sect. 5, the experimental results and analyses are discussed and summarized. In the end, we conclude this paper and propose future works in Sect. 6.

2. Related Work

In this section, we review relevant studies and concepts of recommend system, heterogeneous information network recommend system, and attacks to recommend system.

2.1 Recommend System

The recommend system alleviates the users' information-screening dilemma, thence it is widely deployed in many web services. Figure 1 shows the workflow of the recommend system. This process can be described as: 1) Users' behavior information is input into the recommend algorithm. 2) Recommend system trains model 3) Recommend

Fig. 1 The workflow of recommend system.

Fig. 2 An instance of HIN.

system predicts the product score or selects the Top-K products. 4) Returning results to users. According to the objective of recommendation, RS can be divided into score prediction or Top-K recommend system. Classifying from the recommend algorithm, RS can be divided into collaborative filtering, content-based, model-based, and hybrid recommend system: 1) The key idea of collaborative filtering RS is the similar users [15] or items [16] may have the same preferences. 2) Model-based RS. This kind of RS means using SVD [17], probability matrix decomposition [18], etc. to solve recommend problems. 3) Content-based RS analyzes the attributes of products that users have purchased or viewed, and then pushes the items with the highest similarity to users [19]. 4) The hybrid RS combines different methods to achieve the purpose of improving its recommend effect.

2.2 Heterogeneous Information Network Recommend System

As a newly emerging direction, heterogeneous information networks abstract complex real data more naturally. Like the literature data, it includes different types of objects such as author, paper, conference, etc., and various relationships between objects. Figure 2 (a) shows the HIN formed by the DBLP papers, and Fig. 2 (b) describes the network structure of HIN. We define an information network $G = (V, E, \varphi, \psi)$ consisting of object type mapping function $\varphi : V \rightarrow A$ and a link-type mapping function $\psi : E \rightarrow R$. Each object v in object set V corresponds to a type a in type set A . Similarly, each link e relates with a relationship r . HIN is one type of information network, where the number of type set $|A|$ or relationship set $|R|$ is more than 1. In recent years, some researchers realized the superiority of HIN recommend system. Shi et al. [3] established a weighted heterogeneous information network and designed a meta-path based

collaborative filtering model for recommend system. Works [20]–[22] evaluated the rating prediction based on similarity under meta-paths and matrix factorization.

2.3 Attacks to Recommend System

The mainstream attack method of RS is shilling attack, which is summarized in two steps: 1) Fake users craft bogus ratings and add them to training data. 2) RS learns misleading information from training data and produces the incorrect results subsequently. Currently, the shilling attack is divided into several representative methods like random attack [7], average attack [7], popular attack [23], etc. Subsequent researchers put forward some new injecting attack methods by combining different simple attack schemes. A typical combination is called bandwagon attack, which include bandwagon-average attack [24] and bandwagon-random attack [25]. More recent attacks [9], [10] generated fake behaviors that have been optimized to the specific recommend system. Specifically, Li et al. [9] proposed an attack against the matrix factorization RS, determined the injection by calculating the approximate gradient of RS. Fang et al. [10] proposed a gradient attack method against graph-based RS.

3. Preliminary

In this section, we introduce some preliminaries about the victim model.

3.1 Victim Model

In this work, we focus on a novel HIN recommend system based on network representation, which is called HERec [13], as the victim model. As Fig. 3 shows, HERec

contains the node embedding module and matrix factorization (MF) recommender. We define a heterogeneous information network $G = \{V, E\}$. Multiple types of objects are contained with HIN, such as user (U), movie (M), director (D) and so on. Different types of links between objects represent various relations. The link between two users means friendship, while a user-movie link indicates the rating relation. Recommender mainly focuses on user and item in heterogeneous graphs, which defined as user set $U \subseteq V$, and item set $I \subseteq V$. Triad $(u, i, r_{u,i})$ represents the user $u \in U$ gives one item $i \in I$ a score $r_{u,i}$. HERec uses triad set R as training data, and goals to predict unobserved score $r_{u,i}$.

3.1.1 Node Embedding

Node embedding aims to learn a low-dimensional representation $e_v \in \mathbb{R}^d$ for each node $v \in V$ in HIN. This process is composed of three parts: 1) Using the meta-path to generate node sequences based on random walk. 2) Filtering the relay node on heterogeneous subnets under different semantics and calculating the embeddings with Node2vec. 3) Fusing each embedding, then getting the rich semantic embedding representation of each node. As HERec focuses on user and item, it's only necessary to find the rich semantic representations of users e^U and items e^I .

$$P(n_{t+1} | n_t, A_t) = \begin{cases} \frac{1}{|N_{n_t}^{A_{t+1}}|} (n_t, n_{t+1}) \in R; \\ 0, \text{otherwise}, \end{cases} \quad (1)$$

$$\max_f \sum_{u \in V} \log \Pr(N_u | f(u)), \quad (2)$$

Equation (1) describes the process of capturing high-level semantic information [13], where $G = \{V, E\}$ is a HIN and $\rho : A_1 \xrightarrow{R} \dots \xrightarrow{R_t} A_{t+1} \xrightarrow{R_{t+1}} \dots \xrightarrow{R_l} A_{l+1}$ means one meta-path. n_t is the t node in meta-path with label A_t . $N_{n_t}^{A_{t+1}}$ is

Fig. 3 The schematic diagram of HERec recommend model structure. (a) indicates a heterogeneous information network containing multiple information. The example here is a heterogeneous information network composed of movie data. (b) shows the node embedding module which use meta-paths to capture high-level semantic information and express it as a low-latency vector. The node embedding module is composed of (b1) Meta-path random walk, (b2) Filtering, and (b3) Embedding fusion. This part outputs the rich semantic embeddings of users and items after aggregation; (c) shows the recommendation base on matrix factorization. MF recommender predicts user's scoring for products, which is improved by combining with rich semantic representations.

Fig. 4 An example of meta-path filtering. This part first does random walk with pre-defined meta-paths, and then percolates heterogeneous nodes to retain users and commodity nodes.

the neighbor set with label A_{t+1} of n_t . Filters are used to retain the users or items with different semantic information (see Fig. 4). Equation (2) shows the optimization of learning node representations. Where $f(\cdot)$ is function that map one node to a d dimensional vector. $N_u \subset V$ are nodes in the same sequence with u .

$$g(\{e_s^{(l_s)}\}) = \begin{cases} \frac{1}{|P_s|} \sum_{l_s=1}^{|P_s|} (M^{(l_s)} e_s^{(l_s)} + b^{(l_s)}), \\ \sum_{l_s=1}^{|P_s|} \omega_s^{(l_s)} (M^{(l_s)} e_s^{(l_s)} + b^{(l_s)}), \quad s \in \{u, i\} \\ \sigma_s(\sum_{l_s=1}^{|P_s|} \omega_s^{(l_s)} \sigma_s(M^{(l_s)} e_s^{(l_s)} + b^{(l_s)})), \end{cases} \quad (3)$$

HERec proposes a general function $g(\cdot)$ to fuse node embeddings $e_s^{l_s}$, $s \in \{u, i\}$ under different meta-paths. Aiming to learn users' personalized preference, HERec discusses three fusion functions (see Eq. (3)) to accomplish integration: 1) Simple linear fusion, gives each meta-path a unified weight. Where $|P_s|$ is the number of user or item related meta-path. 2) Personalized linear fusion, dynamically trains preference parameter $\omega_s^{(l_s)}$ of each user/item to meta-path $l \subset P_s$. 3) Personalized non-linear fusion, which uses non-linear function $\sigma(\cdot)$ to enhance the fusion ability. $M_s^{(l_s)} \in \mathbb{R}^{D_s \times d}$ and $b \in \mathbb{R}^{D_s}$ are the transformation matrix and bias vector w.r.t. the meta-path l_s , respectively.

3.1.2 MF Recommender

The user's expected score is calculated based on the matrix factorization model. HERec finds a new optimized calculator incorporating rich-semantic embeddings $e_u^{(U)}$ and $e_i^{(I)}$ into the traditional score calculation $\hat{r}_{u,i} = x_u^T \cdot y_i$, seen in Eq. (4).

$$\hat{r}_{u,i} = x_u^T \cdot y_i + \alpha \cdot e_e^{(U)^T} \cdot \gamma_i^{(I)} + \beta \cdot \gamma_u^{(U)^T} \cdot e_i^{(I)} \quad (4)$$

Representations $x_u \in \mathbb{R}^D$ and $y_i \in \mathbb{R}^D$ are latent feature vectors of user and item. $\gamma_i^{(I)}$ and $\gamma_u^{(U)}$ respectively denote user-specific and item-specific latent factors to pair with the rich-semantic embeddings. α and β are tuning parameters to integrate the three terms [13].

3.2 Attack Circumstance

3.2.1 Attackers' Goal

In this work, attackers try to influence the scoring of a specific product by constructing a reasonable injecting attack method. In particular, the purpose of e-commerce recommend system attacks usually is to increase the sensitometry of a particular product. They hire some fake users to give false scores to the target item, so as to improve the scores of other users on the target item.

3.2.2 Attackers' Capability

This article assumes that the attacker performs a white-box attack [26]. The white-box attackers know the details of the victim model, like the training parameters and training data. It is assumed that the attacker has the ability to inject any number of false nodes into the training data, and each false node can inject a specific score into any commodity.

3.2.3 Attack Restrictions

Since injecting attack needs to inject fake users and ratings into the training data. The implementation method of shilling attack in real e-commerce scenarios is that the seller hires a number of users and designates some products for purchase. The cost paid by the seller is directly proportional to the number of people hired and the amount of false reviews injected. Therefore, hiring as few false users as possible and reducing the amount of false ratings can reduce the cost of employer. Meanwhile, injecting too many fake evaluations lead the graph structure abnormally and is easily detected by the system. Therefore, it is necessary to limit the number of evaluations injected to upgrade the attack success rate.

4. Semantic Shilling Attack

In this section, we introduce the detail of the Semantic Shilling Attack.

4.1 Attack Model

We assume that the goal of the attacker is improving the unobserved ratings of targets as much as possible. Given the original evaluation matrix M , the user set U , the item set I , and a target item $i_t \in I$. The HERec model F trained by injected evaluation matrix $\tilde{M}$ and calculates the expected scoring matrix $\tilde{M}$. We aim to find the $\tilde{M}$, and maximum the predicted score matrix $\tilde{M}_t$ of each user $\tilde{u}_t$ who not bought the target product i_t in test target item set $\tilde{U}_t$, $\tilde{U}_t \subseteq U$. This process is described as the following optimization problem.

$$\max_{\tilde{M}} \tilde{M}_t = \sum_{\tilde{u}_t \in \tilde{U}_t} \tilde{M}_{\tilde{u}_t, i_t} \quad (5)$$

$$\tilde{\theta} = \arg \min F(\hat{M}) \quad (6)$$

$$s.t. |\hat{U}| \leq N, |\hat{H}_{\hat{u}}| \leq n, \hat{r}_{\hat{u},i} \in \{1, 2, \dots, r_m\} \quad (7)$$

where $\tilde{M}_{\hat{u},i_t}$ is the scoring matrix of $\hat{u}_t$ to i_t . $\hat{U}$ is the fake user set and $\hat{H}_{\hat{u}}$ means each fake user's injected rating behavior set. We limit the number of fake user and each fake user's scoring behavior as $|\hat{U}| \leq N, |\hat{H}_{\hat{u}}| \leq n$.

According above reasoning, finding $\hat{M}$ can be transformed into the problem of search injected rating set $\hat{H}$ which is composed of each fake user's injected rating behavior set $\hat{H}_{\hat{u}} = \{(\hat{u}, i, \hat{r}_{\hat{u},i})\}, i \in I$. Where $\hat{r}_{\hat{u},i}$ is a fake score. It should be noted that $\hat{r}_{\hat{u},i}$ is a selected value from the original score set $R = \{1, 2, \dots, r_m\}$, where r_m is the maximum score value.

4.2 Semantic Shilling Attack

HIN extracts the high-level semantic information based on meta-paths and aggregates it into representations of users and items. Adding scoring behaviors to the original scoring data will generate corresponding high-level semantic information simultaneously. According to principle of HERec, fake semantics will be expressed into the rich semantic embeddings no matter which aggregation HERec adopts. Furthermore, compared with the bipartite graph, misleading information can be transmitted to more users/items through high-level semantics. Attackers could indirectly affect the target items by semantic-related items. However, the contribution of user and item embedding changes to the overall scoring are different. Beyond that, we analyze the proportion of three parts in Eq. (4), and find that the semantic embeddings of items have stronger relevance. Thus, we pay close attention to semantic features related to items. Based on those, we propose a heuristic method to solve the attack optimization problem.

We first filter out the set of neighbor nodes I_l^* , $l \in P_I$ that are semantically related to target item under different meta-path l . Noticed, P_I is the set of meta-paths associated with item. After that, we divide the score set $R = \{1, 2, \dots, r_m\}$ into $|R|$ score intervals $A = \{(0, 1), (1, 2), \dots, (r_{m-1}, r_m)\}$ with preset size. Under different meta-paths, each i_l^* pushed into different intervals based on the average score $\bar{r}$. For each fake user, we build the injecting triple and add it to $\hat{H}$ as Algorithm 1.

Algorithm 2 shows the process of selecting filling items for each fake user. We first add triple $(\hat{u}, i_t, r_m)$ to the injected rating behavior set $\hat{H}_{\hat{u}}$. Then we randomly select value r' according to the normal distribution $N(\mu, \sigma^2)$ of the original score matrix, where μ and σ^2 are average and variance of all scores, respectively. Noticed, random value based on the normal distribution is consequent. In fact, the user's evaluation for items is discrete. Here we round continuous value to the nearest integer, represented by function $\psi: r' \rightarrow \hat{r}$, $\hat{r} \in R$. Finally, filling item i_n is selected from I_l^* . Due to meta-paths are previously sorted by importance and thence I_l^* is applied sequentially. In particular, the fake user's selection of neighbor items with different semantics will affect

Algorithm 1: Semantic Shilling Attack

Input: original evaluation matrix M , fake user amount n , number of fake ratings N , target i_t , sorted item meta-paths set P_I .

Output: Fake evaluation behavior set $\hat{H}$

```

1 Empty  $\hat{H}$ 
2 Initialize  $A$ 
3 Sorted item by frequency as set  $C$ 
4 for  $l$  in  $P_I$  do
5   Build first-order neighbor set  $I_l^*$  for  $i_t$ , and delete neighbor from  $C$ 
6   Calculate the average score  $\bar{r}$ , Add the node to the interval  $A_k^l$ 
7 end
8 for  $i = 1$  to  $n$  do
9   for  $r = 1$  to  $N$  do
10    Construct Semantic Injection
11     $\hat{H}_{\hat{u}} \leftarrow (\hat{u}, i, \hat{r})$ 
12  end
13   $\hat{H} \leftarrow \hat{H}_{\hat{u}}$ 
14 end
15 Return  $\hat{H}$ 

```

Algorithm 2: Construct Semantic Injection

Input: number of fake ratings N , item meta-paths set P_I , target i_t , sorted set C

Output: injected triad

```

1 if  $n = 1$  then
2    $\hat{r} = 5$ 
3    $i_n = i_t$ 
4 end
5 else
6    $\hat{r} = \text{random select from } N(\mu, \sigma^2)$ 
7   if  $I_l^* \neq \emptyset$  then
8     for  $l$  in  $P_I$  do
9        $i_n = I_l^*[n-1]$ 
10       $I_l^*$  remove  $i_n$ 
11    end
12  end
13 else
14    $i_n = C[n-1]$ 
15 end
16 end
17 return  $(\hat{u}, i_n, \hat{r})$ 

```

the attack cost. For example, in a real e-commerce attack, compared to scoring products from other stores, injecting fake scores to products in the same store can significantly reduce the attackers' cost. Therefore, the priority of selection will change according to attack cost.

One example is cited to illustrate our strategy seen as Fig. 5. For example, i_B and i_C respectively mean the neighbor items of the target items under the meta-path 'UISIU' and 'UITIU'. Average scores of two neighbor items are 3 and 2. When the value obtained by the state distribution is 4, it is given priority to i_B . When it is 3, given priority to i_C . And the remaining scores are given to other popular products.

Intuitively, the essence of HIN fusion high-level

Fig. 5 An example for Semantic Shilling Attack.

semantic recommendation is actually a process of gathering more information. Under the modeling of a HIN, users not only refer to the scoring behavior of those who have purchased the same goods, but also refer to the scoring situation of the same store/type of goods. Therefore, if the attacker intends to increase the total score of the target item, it is necessary to increase the score of the same store/category item within the limited cost. The attack method proposed in this article that fake users give priority to improving the scores of neighbor items, is to achieve the above process. At the same time, adding fake scores based on the average score of items can well circumvent the detection methods that use local and global score distributions.

5. Experiments

Experiments are performed on two real datasets. Comparing with different baseline algorithms, we demonstrate the effectiveness of SSA. The experiments discuss two questions we focus on as below:

RQ1: Compared with bipartite graph models, what new characteristics does HIN show in attack scenarios?

RQ2: Is our method more effective than other shilling attack methods?

5.1 Experimental Setup

5.1.1 Dataset

We adopt two widely used real-world datasets in recommend system, which are named ML-100K [27] and Yelp [13].

ML-100K. ML-100K consists of 100,000 ratings of 1,682 movies by 943 users, which contains information such as movie publication year and genre users' occupation and gender.

Yelp. Yelp records the ratings of 14,282 businesses by 16,239 users. This dataset contains information about businesses' social relationships and attributes (e.g., location, services' categories and other information), as well as users'

Table 1 Datasets statistic analysis

Dataset	#Users	#Items	#Ratings	#Scores	#Sparsity
ML-100K	943	1682	100000	1, 2, ..., 5	93.6%
Yelp	16,239	14,282	198397	1, 2, ..., 5	99.2%

Table 2 Dataset semantic information analysis

Dataset	#Feature	#Numbers	#Meta-path
ML-100K	Movie categories	19	MCaM, UMCaMU,
	Movie release-time	72	MTM, UMTMU,
	User occupation	21	MUM, UMU
Yelp	Business city	47	BCiB, UBCiBU,
	Businesses categories	511	BCaB, UBCaBU,
	User attitudes	11	BUB, UBU

information (e.g., attitude in comments).

Another important common property of RS dataset is sparsity. We follow the definition of sparsity proposed in the literature [5]:

$$\text{Sparsity} = 1 - \frac{\text{number of ratings}}{\text{number of users} \times \text{number of items}} \quad (8)$$

In this work, we discuss how sparsity influences the attacking effect. Simultaneously, meta-paths on ML-100K and Yelp are predefined. To avoid introducing noise, we choose meta-paths which shorter than four steps same as the literature [13]. Statistics and meta-paths of two datasets are shown in Table 1 and Table 2. In different datasets, the specific explanation of semantic information contained in each meta-path is shown in Table 3.

5.1.2 Baseline

We design some comparative tests with several typical shilling attack methods. Each attacker injects N fake users into the original data, and each fake user adds the highest score for the target item and $n - 1$ fillers with score. The difference between each baseline is strategies of generating $n - 1$ filling items and their scores. The SSA is compared with the following models:

Table 3 The semantics and descriptions of different meta-paths in ML-100K and Yelp

Dataset	#Object	#Meta-path	#Semantics and description
ML-100K	User	UMU	Users who rated the same movie
		UMCaMU	Users who rated two movies with the same category
		UMTMU	Users who rated two movies have the same release-time
	Movie	MTM	Movies which have the same release-time
		MCaM	Movies which have the same category
		MUM	Movies reviewed by the same user
Yelp	User	UBU	Users who paid the same business
		UBCiBU	Users who paid two businesses locate in the same city
		UBCaBU	Users who paid two businesses with the same category
	Business	BCiB	Businesses which locate in the same city
		BCaB	Businesses which have the same category
		BUB	Businesses paid by the same user

Random attack (RA). In this attack [7], attackers calculate the normal distribution of the training data scores, and take a random value from the distribution as the fake score. For items, $n - 1$ fillers are randomly selected from all commodities.

Popular attack based Item (PA-I). A popular attack be modified by the method from literature [27]. We choose the top $n - 1$ items from the training data popularity as fillers. We believe that the bigger number of purchasing, the greater popular of this item. The selection of fake scores is the same as the random attack.

Popular attack based Score (PA-S). A variant of a popular attack in literature [27]. The fake score setting is similar as the PA-I, and the strategy of choosing filling users is different. We believe that the total score of an item can comprehensively measure its popularity in terms from the number of purchasers and its average score. Therefore, PA-S selects users with the top $n - 1$ total score as injecting items.

Bandwagon attack (BA). BA attack [28] is a typical combined injecting attack method, which selects filler combined of popular attack based on average scores and random attack. Each fake user selects $n \times 10\%$ items with the highest average score from the training set and randomly selects $n \times 90\%$ items to compose an injected sequence. The fake score is generated from the normal distribution of the user score matrix (same as the random attack).

5.1.3 Metrics

Average promotion score (APS). For RQ1, we use average promote score (APS) to measure the effectiveness of different attacks. It changes from the indicator average promotion ratio in literature [27], and it can reflect the macro attacking effect. The evaluation defined as below.

$$APS = \frac{1}{|T|} \sum_T \tilde{M}_t - M \quad (9)$$

where T is the set of target items. $\tilde{M}_t$ is the predicted total score of each target item after the attack, and M_t is the original predicted total score.

Average promotion ratio (APR). Average promote ratio (APR) [27] is a more fine-grained metric. Using it, we can

distinguish the improvement effect of the model on different target products and can make the more targeted analysis. It is used to measure the effectiveness of different attacks in RQ2. The evaluation is defined as below.

$$APR = \frac{1}{|T|} \sum_T \frac{\tilde{M}_t - M_t}{M_t} \quad (10)$$

5.2 High-Level Semantic Characteristics (RQ1)

We believe that HIN recommender is easier to accept misleading information from attackers based on the recommend principle of HIN, which is compared with the bipartite graph RS. That means incorporating meta-paths will enhance the attacking effect. In order to verify assumptions, a random attack is designed on HERec-pl model (HERec using personalized linear fusion). Referring the amount of injection proposed in literature [10], we respectively inject 10 and 160 fake users into ML-100K and Yelp. Each fake user gives the highest score to 10 target products, and then selects 10 random items and gives each of them a random score. This experiment uses the APS of the target total prediction as evaluation, when recommender incorporating meta-paths gradually.

To ensure accuracy, we calculate the average of repeated attacks. We sequentially add the previously defined meta-paths to the model, and the upgrade of APS as Fig. 6. Specifically, “+Null” means recommender performed without using any meta-path. What interesting is, adding certain meta-path causes the attacking effect shaking. This appears to be a case of meta-paths containing a certain amount of noise. Another phenomenon is adding equivalent semantic information that has the same depth does not significantly improve the attacking effect (such add “UMU” first, then add “MUM”). The reason is considered as meta-paths included both users’ and items’ information during diffusion, so adding the equivalent semantic meta-path realizes the information-enhanced delivery only. In short, although increasing the number of used meta-paths can improve the recommendation effect, but reduces the robustness of the model.

5.3 Attack Performance (RQ2)

Semantic priority. We simulate the attacking intent in the commercial recommend scenario on ML-100K and Yelp. Therefore, the high-level semantic priority of ML-100K is publishing time more than movie type. In Yelp, the priority of city is higher than service categories' priority.

Overall comparison. Table 4 shows the average promote ratio of the three categories sample under different methods

Fig. 6 The upgrade of APS when gradually incorporating meta-paths

and attack sizes. The attack size is indicated as the proportion of fake users to real users. We set three attack sizes: 1%, 2%, and 3%. And we rank three types of item: random, popular and unpopular. Specifically, random items are selected from item list randomly. Due to the difference in sparseness, we define popular and unpopular product according to different datasets, and comprehensively filter them from purchased times and the total score. In ML-100K, popular items are defined as bought more than 80 and score more than 320; unpopular items are bought less than 10 and score less than 20. In Yelp, popular items are purchased more than 35 and fraction exceed 140; unpopular items are less than 4 and score less than 10. We preset the sample size is 15 and calculate the average of multiple rounds.

Our method goes beyond previous shilling attacks in ML-100K and Yelp. In ML-100K, our attack method exceeds the baseline method on unpopular products and random products. For popular items, our attack method basically equal to the best attack. In Yelp, our method is always better than the traditional method. Particularly, to popular items in ML-100K, PA-S gets better results with 2% and 3% attack size. The explanation of this result is when users purchase popular products, the information conveyed by high-level semantically related products may contain a certain amount of noise. And due to popular items have lots of semantic-related items, the influence of semantic attack has a certain degree of attenuation. Therefore, SSA does not achieve the best result.

Impact of attack size. Figure 7 and Fig. 8 show the effect of different attack sizes. In the ML-100K data, the correlation between the attacking effect and attack size is positive. Among, baseline methods maintain a relatively stable growth rate, and the difference in growth rate is not significant. SSA grows faster in random products, then slows down with the attack size increases. For popular products, the effectiveness is lower than other baseline methods at 2% and 3%. For unpopular products, the effect of SSA is much higher than other baselines. In Yelp, SSA is significantly better than other baseline methods under three attack sizes. The random attack method has a large jitter. The possible reason is that the randomness of the random select strategy is too large. If the selection hits the appropriate rating

Table 4 Datasets statistic analysis

Dataset	Attack	Inject size								
		Random item			Popular items			Unpopular items		
		1%	2%	3%	1%	2%	3%	1%	2%	3%
ML-100K	RA	0.20212	0.31880	0.39694	0.05087	0.11363	0.16161	0.17628	0.22910	0.29114
	PA-I	0.24671	0.36254	0.43439	0.05640	0.11848	0.16157	0.27603	0.32533	0.37707
	PA-S	0.24681	0.35598	0.43696	0.05542	0.12122	0.16617	0.27809	0.32358	0.37868
	BA	0.20999	0.30627	0.40567	0.05544	0.11348	0.15278	0.20230	0.22661	0.33165
	SSA	0.35345	0.59895	0.64253	0.06093	0.10360	0.13901	0.56347	0.65938	0.69908
Yelp	RA	0.08652	0.07113	0.07929	0.12132	0.12974	0.11676	0.08243	0.05341	0.05462
	PA-I	0.06180	0.07658	0.08429	0.04686	0.04776	0.04275	0.08402	0.08566	0.12108
	PA-S	0.06007	0.06331	0.06903	0.05980	0.06261	0.06245	0.05891	0.06552	0.07234
	BA	0.07808	0.08477	0.08115	0.06042	0.05957	0.06153	0.10086	0.12158	0.09966
	SSA	0.20194	0.21109	0.19718	0.19931	0.18957	0.17810	0.20576	0.22685	0.21557

Fig. 7 The APR changes with different attack sizes for each category item on ML-100K

Fig. 8 The APR changes with different attack sizes for each category item on Yelp

behaviors, it will have a better performance. In particular, the SSA effect on Yelp no longer increases with the increased attack size. This is because the number of fake users injected is large, and the number of target's semantic-related items is limited. In the end, it enables attacking effectiveness to reach the upper limit after the candidate filling items completely selected.

Compare with different items. In ML-100K, our method is more excellent than other baseline algorithms and can get the result twice better than random strategy. Among popular items, our method has basically the similar results as others. For unpopular items, our method obtains result several times better than other methods. In Yelp, SSA has always maintained a high level of effect. For random commodities, baselines are nearly staying the same. For popular items, RA is better than other baseline methods, and the effect of SSA is nearly twice bigger than RA's. For unpopular items, baselines have a large jitter, and our method turns out to be about twice better than others. Note that including both popular and unpopular items, the sampling of random can better reflect the overall consequence of the attack. And for unpopular items, our attacking method has better effectiveness. We deduce the reason is the information transmitted by semantic-related neighbors of unpopular targets is more related to itself than popular commodities. Furthermore, pair of popular products is more likely to become semantic-related neighbors, which caused fakers' intensity of misleading transmission reduced. Therefore, our attack perform relatively better on unpopular products. In addition, the attack influence largely depends on the sparsity of the data-self. We speculate that the relatively low sparsity makes popular items easier to introduce noise semantics. Thence, SSA is unable to achieve good result for

popular items on ML-100K.

6. Conclusion

In this work, we propose a shilling attack method based on semantic information against the HIN recommend system. Our research prove that the difficulty of misleading information disseminating on HIN is inversely proportional to the number of meta-paths with different depths. We choose filling users from semantically related neighbors to ensure that fake scoring information can be transmitted to the target as much as possible. We obtain good results with this simple method in two real datasets compared with four baseline methods. However, when comparing our results to other studies, it must be pointed out there are some limitations of our strategy. A major source of limitation is due to attackers' knowledge, which hard to reach in reality.

Our future work includes: 1) Exploring more practical HIN recommend system attacks. Particularly, we need to verify the scalability of SSA on latest HIN recommend systems that use meta-paths or other methods to obtain the high-level semantics. 2) Optimizing semantic selection strategies instead of heuristics. The latest graph deep learning attack work, which uses gradients to solve attack samples, has achieved better results than heuristic attacks. Therefore, future HIN recommend system attack models should try to use gradients to get fake scoring behaviors. 3) Finding ways to improve the robustness of the HIN recommend system. With the development of deep learning research, model defensive research has developed rapidly. And compared with attack research, defensive research has more practical significance.

References

- [1] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Knowl. Data Eng.*, vol.17, no.6, pp.734–749, 2005.
- [2] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM Computing Surveys (CSUR)*, vol.52, no.1, pp.1–38, 2019.
- [3] C. Shi, Z. Zhang, P. Luo, P.S. Yu, Y. Yue, and B. Wu, "Semantic path based personalized recommendation on weighted heterogeneous information networks," *Proc. 24th ACM International Conference on Information and Knowledge Management*, pp.453–462, 2015.
- [4] Z. Wang, H. Liu, Y. Du, Z. Wu, and X. Zhang, "Unified embedding model over heterogeneous information network for personalized recommendation," *IJCAI*, pp.3813–3819, 2019.
- [5] S. Fan, J. Zhu, X. Han, C. Shi, L. Hu, B. Ma, and Y. Li, "Meta-path-guided heterogeneous graph neural network for intent recommendation," *Proc. 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp.2478–2486, 2019.
- [6] M. Si and Q. Li, "Shilling attacks against collaborative recommender systems: a review," *Artificial Intelligence Review*, vol.53, no.1, pp.291–319, 2020.
- [7] S.K. Lam and J. Riedl, "Shilling recommender systems for fun and profit," *Proc. 13th international conference on World Wide Web*, pp.393–402, 2004.
- [8] M.P. O'Mahony, *Towards robust and efficient automated collaborative filtering*, Ph.D. thesis, Citeseer, 2004.
- [9] B. Li, Y. Wang, A. Singh, and Y. Vorobeychik, "Data poisoning attacks on factorization-based collaborative filtering," *arXiv preprint arXiv:1608.08182*, 2016.
- [10] M. Fang, G. Yang, N.Z. Gong, and J. Liu, "Poisoning attacks to graph-based recommender systems," *Proc. 34th Annual Computer Security Applications Conference*, pp.381–392, 2018.
- [11] Y. Deldjoo, T. Di Noia, and F.A. Merra, "Assessing the impact of a user-item collaborative attack on class of users," *arXiv preprint arXiv:1908.07968*, 2019.
- [12] K. Chen, P.P. Chan, F. Zhang, and Q. Li, "Shilling attack based on item popularity and rated item correlation against collaborative filtering," *International Journal of Machine Learning and Cybernetics*, vol.10, no.7, pp.1833–1845, 2019.
- [13] C. Shi, B. Hu, W.X. Zhao, and S.Y. Philip, "Heterogeneous information network embedding for recommendation," *IEEE Trans. Knowl. Data Eng.*, vol.31, no.2, pp.357–370, 2018.
- [14] Y. Sun, J. Han, X. Yan, P.S. Yu, and T. Wu, "Pathsim: Meta path-based top-k similarity search in heterogeneous information networks," *Proc. VLDB Endowment*, vol.4, no.11, pp.992–1003, 2011.
- [15] J.S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," *arXiv preprint arXiv:1301.7363*, 2013.
- [16] M. Deshpande and G. Karypis, "Item-based top-n recommendation algorithms," *ACM Transactions on Information Systems (TOIS)*, vol.22, no.1, pp.143–177, 2004.
- [17] H. Polat and W. Du, "Svd-based collaborative filtering with privacy," *Proc. 2005 ACM symposium on Applied computing*, pp.791–795, 2005.
- [18] F. Ortega, A. Hernando, J. Bobadilla, and J.H. Kang, "Recommending items to group of users using matrix factorization based collaborative filtering," *Information Sciences*, vol.345, pp.313–324, 2016.
- [19] Z.h. Xu and B.y. Wang, "Collaborative filtering algorithm based on item complex similarity," *Application Research of Computers*, p.02, 2014.
- [20] C. Shi, J. Liu, F. Zhuang, S.Y. Philip, and B. Wu, "Integrating heterogeneous information via flexible regularization framework for recommendation," *Knowledge and Information Systems*, vol.49, no.3, pp.835–859, 2016.
- [21] J. Zheng, J. Liu, C. Shi, F. Zhuang, J. Li, and B. Wu, "Recommendation in heterogeneous information network via dual similarity regularization," *International Journal of Data Science and Analytics*, vol.3, no.1, pp.35–48, 2017.
- [22] J. Zheng, J. Liu, C. Shi, F. Zhuang, J. Li, and B. Wu, "Dual similarity regularization for recommendation," *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp.542–554, Springer, 2016.
- [23] N.J. Hurley, M.P. O'Mahony, and G.C. Silvestre, "Attacking recommender systems: A cost-benefit analysis," *IEEE Intell. Syst.*, vol.22, no.3, pp.64–68, 2007.
- [24] Z. Wu, J. Wu, J. Cao, and D. Tao, "Hysad: A semi-supervised hybrid shilling attack detector for trustworthy product recommendation," *Proc. 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp.985–993, 2012.
- [25] Z. Yang, Z. Cai, and X. Guan, "Estimating user behavior toward detecting anomalous ratings in rating systems," *Knowledge-Based Systems*, vol.111, pp.144–158, 2016.
- [26] H. Xu, Y. Ma, H.-C. Liu, D. Deb, H. Liu, J.-L. Tang, and A.K. Jain, "Adversarial attacks and defenses in images, graphs and text: A review," *International Journal of Automation and Computing*, vol.17, no.2, pp.151–178, 2020.
- [27] Q. Zhou, Y. Ren, T. Xia, L. Yuan, and L. Chen, "Data poisoning attacks on graph convolutional matrix completion," *International Conference on Algorithms and Architectures for Parallel Processing*, pp.427–439, Springer, 2019.
- [28] B. Mobasher, R. Burke, R. Bhaumik, and C. Williams, "Toward trustworthy recommender systems: An analysis of attack models and algorithm robustness," *ACM Transactions on Internet Technology (TOIT)*, vol.7, no.4, pp.23–es, 2007.

CSS2009 Student Paper Award and AINA2011 Best Student paper Award.

Yizhi Ren received the Ph.D in Computer software and theory from Dalian University of Technology, China in 2011. From 2008 to 2010, he was a research fellow at Kyushu University, Japan. He is a professor with School of Cyberspace, Hangzhou Dianzi University, China. His research interests include: big data content security, artificial intelligence security, etc. Dr. REN has published over 70 research papers in refereed journals and conferences. He won IEEE Trustcom2018 Best Paper Award, CSS2009 Student Paper Award and AINA2011 Best Student paper Award.

Zelong Li received the B.E. in Information Security from Hangzhou Dianzi University, China in 2019. He currently pursuing the master degree in the School of Cyberspace, Hangzhou Dianzi University. His research interests include: big data security, artificial intelligence security, network security, etc.

Lifeng Yuan received the Ph.D in Computer software and theory from Dalian University of Technology, China in 2017. He is a lecturer with School of Cyberspace, Hangzhou Dianzi University, China. His research interests include: information security, secret sharing and information hiding.

Zhen Zhang received his M.S. degree in Computer Science from Zhejiang University, Hangzhou, China, in 2005. He studied as a visiting scholar in University of Kentucky, USA, from 2013 to 2014. He has been working in Hangzhou Dianzi University as a faculty since 2000 and Now he is an Associate Professor of the School of Cyberspace in HDU. His current research interests include information security, information hiding, confidential technology, etc.

Chunhua Su received his PhD of computer science from Faculty of Engineering, Kyushu University, Japan in 2009. He is currently working as Senior Associate Professor in Division of Computer Science, University of Aizu. He has worked as a research scientist in Cryptography and Security Department of the Institute for Infocomm Research, Singapore from 2011–2013. From 2013–2016, he has worked as an Assistant professor in School of Information Science, Japan Advanced Institute of Science and Technology.

From 2016–2017, he worked as Assistant Professor in Graduate School of Engineering, Osaka University. His research interests include cryptanalysis, cryptographic protocols, privacy-preserving technologies in data mining and IoT security and privacy.

Yujuan Wang is a lecturer with School of Cyberspace, Hangzhou Dianzi University, China. Her research interests include network security and data privacy and security.

Guohua Wu received the B.S. degree from Shandong University of Technology, Jinan, China, in 1992, and M.S. degree from National Institute of Metrology, Beijing, China, in 1995, and Ph.D. Degree from Zhejiang University, Hangzhou, China, in 1998. From 1998 to 2000, he was a lecture with the Department of Biomedical Engineering, Zhejiang University, and became an associate professor in 2001. From 2002, he was introduced to the department of computer science and engineering at

Hangzhou Dianzi University, and was appointed as a professor in 2009. His currently research interests include information system, model driven architecture, data mining.