

Privacy and Security Issues in Recommender Systems

HU Bing

The Chinese University of Hong Kong
Hong Kong SAR, China
1155151103@link.cuhk.edu.hk

LENG Heng

The Chinese University of Hong Kong
Hong Kong SAR, China
1155152543@link.cuhk.edu.hk

ABSTRACT

Recommender system plays a more and more important role in our daily life. When we use Taobao or TikTok, it seems like the system always knows what we want to buy or what we want to watch. It is recommender system that make such magic. But behind this kind of magic, there are still some privacy and security issues that might affect our lives. In this paper, we are going to first introduce you the concept of recommender system and some common recommendation algorithms. And then to the most commonly used recommendation algorithm, collaborative filtering algorithms, we discuss the privacy and security issues related. In the privacy issues section, we summarized two common privacy protection methods in current recommender systems, encryption and distributed storage. In the security issues section, we introduced the most common attack for collaborative filtering recommender systems, the shilling attack, together with its common models and detection methods. Finally, we gave a conclusion and looked to the future recommender systems.

KEYWORDS

recommender system, privacy preserving, security, shilling attack, collaborative filtering

ACM Reference Format:

HU Bing and LENG Heng. 2020. Privacy and Security Issues in Recommender Systems. In *Proceedings of ACM Conference (Conference'17)*. ACM, New York, NY, USA, 7 pages. <https://doi.org/10.1145/1122445.1122456>

1 INTRODUCTION

The emergence and popularization of the Internet has brought a lot of information to users and met their needs for information in the information age. However, with the rapid development of the network, the amount of online information has increased significantly, which makes users unable to obtain the part of information that is really useful to themselves when facing a large amount of information, and the efficiency of using information is reduced, which is the so-called information overload problem.

A very potential solution to the problem of information overload is recommender system, which is a personalized system that recommends the information and products that users are interested in. The recommender system has three important modules:

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

Conference'17, July 2017, Washington, DC, USA

© 2020 Association for Computing Machinery.

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...\$15.00

<https://doi.org/10.1145/1122445.1122456>

user modeling module, recommendation object modeling module and recommendation algorithm module. The recommender system matches the requirement in the user model with the features in the recommended object model. At the same time, the corresponding recommendation algorithm is used to calculate and filter the recommended object that the user may be interested in, and then recommend it to the user. Compared with search engine, recommender system can find user's interest point by studying user's preference, and then guide user to find his requirement. A good recommender system can not only provide personalized services for users, but also establish a close relationship with users, so that users can rely on the recommendation.

The emergence of recommender systems provides users with a powerful tool to solve the problem of information overload. But recommendation systems are also facing severe privacy and security issues. Due to the large amount of user information obtained by the collaborative filtering algorithm, the user's privacy is threatened. And due to the openness and user participation of the Web site, the recommender system is vulnerable to data injection attacks and the recommendation results can be disturbed.

With the increasingly fierce competition in e-commerce, the security of the recommendation system will become more and more important. In this article, we will discuss the privacy issues and security issues we care about in the recommender system, explore mainstream solutions, and finally give a prospect for the future recommendation system.

2 CURRENT RECOMMENDATION ALGORITHMS

Before introducing privacy and security issues, let's take a look at the current mainstream recommendation algorithms. Recommendation algorithms is the most important part of the whole recommender system, and largely decides the type of recommender system and the performance. There are some recommendation algorithms that are currently used, including collaborative filtering recommendation (CF), content-based recommendation (CBR), knowledge-based recommendation and hybrid recommendation. The most basic are content-based and collaborative filtering recommendation. For the collaborative filtering recommendation, the key is to find whether the user's purchase behavior is similar to that of other users, such as the purchase behavior of the user, the collaborative selection model, and so on. This model can be used to predict which items users are likely to be interested in (or how interested they are). As for the content-based recommendation, similar items are recommended by with similar properties by using features of related items. The two methods often combine with each other (refer to hybrid recommendation system). For example, Music software such as Spotify generally uses collaborative filtering recommendation system to recommend songs that users

may like. It matches similar users by analyzing users' preferences and frequency of listening songs in the past, and then recommends other music that is not in the user's music library but often played by other similar users. Another way is to classify each track, and then recommend songs of similar categories according to users' preferences, and then revise them continuously.

CBR is based on the content of the items that users have selected. It does not need to rely on the user's evaluation of the item. Items are defined by the attributes of relevant features. Based on users' evaluation, the system learns their interests and investigates the matching degree of user and items to be predicted. The user's data model depends on the machine learning methods used, such as decision tree, neural network and vector based representation. Content based user profile is the historical data of users, and the user profile model may change together with user preferences. But the disadvantage is that, to help the content to be easily extracted into meaningful features, it should be guaranteed that the feature should be well structured and the user's interest must be expressed in the form of content features. Therefore, it's not as widely used as CF recommender system is.

Collaborative filtering (CF) is one of the most successful technique in recommender system. The basic idea of CF is that, if two users have similar preferences for one item, then each user may be interested in another item. This is easy to understand since in our daily life we usually make decisions according to our friends' recommendation. The more information each user provides about his or her interests, the more meaningful the proposal result will be. CF-based recommender system is automatic, which means that the recommendation results that users got do not require users to search for information or fill in some research forms. Instead, the system obtains those information implicitly from user behavior records. Another advantage is that there is no specific restriction about the recommendation items, while the content-based recommender system needs to do feature analysis about the recommendation items. At the same time, there are already a large amounts of user behavior data obtained from the past social network researches, making a solid research foundation and broad prospects.

Generally, the collaborative filtering algorithm based on users can be divided into three steps:

- * building user profiles
- * computing between users
- * predict the score of the items unrated, and generate the top n recommendation items by methods like KNN (K-nearest-neighbors).

3 PRIVACY ISSUES

3.1 Concepts

Due to the limited local computing resources of mobile users, the establishment of the prediction model of the recommender system and the calculation of the recommender results are usually outsourced to a recommender server with sufficient storage and computing resources. However, the recommendation server generally works under a semi-trusted or malicious model. The former refers to that the recommender server honestly executes in accordance with the provisions of the agreement, and at the same time obtains the secret information about the user to the greatest extent

through interaction with the user. The latter refers to that the recommender server can destroy the implementation of the agreement through any behavior. Therefore, the privacy preservation of the recommender system faces a dilemma: on the one hand, in order to improve the accuracy and usability of the recommendation results, the system needs to extract the user's relevant historical data information (user attributes, item attributes, ratings, etc.) as the training set of the prediction model on a large scale and with high accuracy. On the other hand, the larger and more specific the user's historical data is, the greater the risk of privacy exposure and the lower the efficiency of the recommender system (user-side storage overhead, computing overhead and communication overhead) will be. Therefore, how to solve the problem of efficient privacy protection in the recommender system, is a issue that needs to be solved urgently and has important theoretical significance and social value.

3.2 Privacy-preserving Techniques

Several techniques have been proposed to preserve the privacy of users in recommender systems. Perturbing users' ratings, using cryptographic tools such as homomorphic cryptography, and storing users' profiles in a distributed manner are the main categories for privacy preservation in collaborative filtering systems.[10]

3.2.1 cryptographic methods. Polat and Du [9] propose a randomized perturbation technique to protect privacy in CF systems. By adding random noise to users' ratings, the central server can not derive the users' real ratings. The challenge is to find a perturbation algorithm with the smallest error. If the server cannot estimate the real ratings users assigned to the items, the users could enjoy privacy in a high level.

To hide the operations of the recommender system, Canny [4] proposed the idea of using homomorphic cryptography in the public server. With the hope of getting more valuable recommendation results, users create communities and each of them searches for recommendations from the most appropriate community, instead of searching for help from users who have similar profiles. Each community of users compute a public aggregate of profiles and individuals' profiles are hidden. Although performed by the server, homomorphic cryptography hide the aggregation operation from the server. Users' participation in the distributed system was assumed to happen but might not be the case in reality. Moreover, the implementation of such a cryptographic strategy is difficult to achieve, due to the status of the current usage of cryptographic systems in the Internet.

3.2.2 distributed storage. Another choice is to store users' profiles on their own side. Recommender system is running in a distributed manner without relying on servers. Berkovsky et al. [1] propose a distributed P2P system to prevent users' profiles from storing on a single server. Although this scheme eliminates the main privacy-threaten source, it requires high cooperation among users to get meaningful recommendations. Every user pays the price whether he interests in privacy preserving or not.

Lathia et al. [6] propose a concordance method to evaluate the similarity between two users in a distributed system. This method avoid revealing users' actual profiles to each other. A temporal profile is randomly generated and shared between two users. Both

	Item1	Item2	Item3	Item4	Item5	Item6	Correlation with Alice
Alice	5	2	3	3		?	
User1	2		4		4	1	-1.00
User2	3	1	3		1	2	0.76
User3	4	2	3	1		1	0.72
User4	3	3	2	1	3	1	0.21
User5		3		1	2		-1.00
User6	4	3		3	3	2	0.94
User7		5		1	5	1	-1.00
Attack1	5		3		2	5	1.00
Attack2	5	1	4		2	5	0.89
Attack3	5	2	2	2		5	0.93
Correlation with Item6	0.85	-0.55	0.00	0.48	-0.59		

Figure 1: an example of a user-based collaborative recommender system being affected by shilling attack.[8]

of the users compute the number of concordant, discordant and tied pairs of ratings between their own profile and the temporal profile. By exchanging the results, they are able to evaluate the similarity between themselves. Because of that, they can keep the rated items as well as the rating values private. In this method, users need to reveal their profiles to generate recommendations. Hence, this method provides privacy only for calculating similarity, not for a whole CF system.

4 SECURITY ISSUES: SHILLING ATTACK

4.1 Concepts

User profile refers to the personal data used to record user preferences and interests in the recommender system. Since the recommender system generates recommender lists based on similar users or similar products, malicious users would be able to change the recommendation results by injecting fabricated user profile into the recommender system. This method of injecting fabricated user profile is called a shilling attack. [2]

4.1.1 Purpose of the attack. One of the purposes of the shilling attack is affecting the recommendation frequency of the target item. This kind of effect could be either increasing (which is called push attack) or reducing (which is called nuke attack). Figure 1 shows an example of a user-based collaborative recommender system being affected by shilling attack. The purpose of this profile injection attack is to increase the rating of item6. Before injecting the fabricated attack 1-3, the user most familiar with Alice is user3, but the rating of item6 for user3 is 0; after injecting attack 1-3, the user most familiar with Alice became attack1. Since that all the ratings of item6 for attack 1-3 are 1.0, the result of the rating of item6 for Alice will see a substantial increase. Another possible purpose of shilling attack is to disrupt the recommendation accuracy of the

entire recommender system, so that users lose trust in the system, and eventually stop using the recommender system.

4.1.2 Knowledge needed. Before carrying out a shilling attack, it is necessary to know the relevant knowledge of the victim recommender system to a certain extent, such as item information, user information, scoring information, and the recommendation algorithm used. Generally speaking, a further understanding of the recommender system, such as the sparsity of the rating, the distribution of the rating, and the parameters of the recommendation algorithm, will help to select which attack algorithm to use, adjust the parameters and reduce the possibility of being detected.

4.1.3 Cost of the attack. The cost of the shilling attack includes knowledge cost and execution cost. The cost of knowledge refers to the acquisition of relevant information about the recommender system and its users required to conduct an attack. The more relevant information required for an attack, the higher the cost of the attack. Execution cost refers to the effort spent interacting with the system in order to submit the profile information needed for the attack. An attack that requires only a small amount of profiles is more practical to mount and more difficult to detect.

4.2 Attack Model

Attack model refers to the method of constructing an attack profile on the basis of related knowledge of the recommender system and its database, products and users. An attack profile is an m-dimensional vector. Where m is the number of items in the recommender system. An attack profile is divided into three parts: filled items, unrated items and target items. Unrated items refer to items without a rating. Assuming that the highest rating value of the recommended system is R_{max} and the lowest rating value is R_{min} , the predetermined value R_m of the target item will be R_{max} when

mounting a push attack, and R_{min} when nuke attack[5]. Figure 2 is a general form of a push attack profile. Currently, the main attack models are as follows:

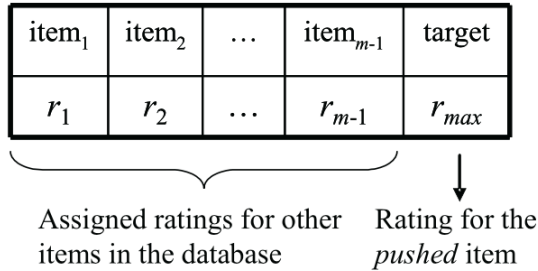


Figure 2: The general form of a push attack profile.

4.2.1 random attack. [5] Target item is given the predetermined value and filled items are given random values within the rating scale with a distribution centered around the average value of all user ratings in all items. Although the attacker does not know the distribution of the ratings, he can estimate this value relatively easily, such as by observing the ratings of other users, or obtaining a sample of the ratings. This kind of attack requires less knowledge, but because the items that need to be filled during the attack are relatively large, its execution cost is high. Experiments show that this attack is less efficient.

4.2.2 average attack. [5] This model requires the attacker to know the average rating of each item. In fact, many recommendation systems are happy to tell users this information. In addition, the attacker may also learn the aggregated data in other ways. For example, some websites or reviews will often publish the average rating of a movie. The difference between the average attack and the random attack model in the attack profile is that the rating of each filled item of the former is the average value; the others are the same. In terms of the effect of the attack, the mean attack model is more effective for the collaborative filtering algorithm based on neighbor users, but it is less effective for the item-based collaborative filtering algorithm.

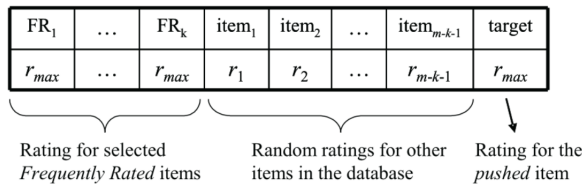


Figure 3: A bandwagon attack profile.

4.2.3 bandwagon attack. [3] Its basic idea is based on Zip's law, that is, a few items can attract the attention of most people. In the recommender system, the attacker selects those popular or best-selling items (which account for a small part of all items) as the selected items of the attack profile, assigns them the maximum rating value, and assigns the target item the predetermined value. In this way, it is very likely that the attack profile is similar to many users. This will help to achieve the purpose of push or nuclear attack. In a bandwagon attack, the filled items are divided into two parts: selected filling items and unselected filling items. The selected filling items are those popular items, and they are all assigned the maximum rating value; the unselected filling items are those non-popular items, and they are rated as the filled items in the random attack model. The target item will be set the maximum rating value (push attack) or the minimum rating value (nuke attack) according to the purpose of the attack. Figure 3 is a bandwagon attack profile.

4.2.4 segmented attack. Because the three models mentioned above are not aimed at certain types of users, it is very likely that the recommended target items may be impossible to be purchased by some users during push attacks. The goal of the segmented attack model is to recommend target items for a specific user group. For example, a certain writer wrote a children's book, and he hopes to recommend his book to users who like to read children's books, such as purchasers of Harry Potter, rather than C++ or buyers of motorcycle repairs. The overview of the segmented attack model can also be illustrated in Figure 4. The attacker first needs to know which items are both similar to his target item and are more popular. These items are selected and assigned the maximum rating value; target items are assigned a predetermined value; non-selected filler items are assigned the minimum rating value. Segmented attack is more effective for item-based collaborative filtering algorithms.

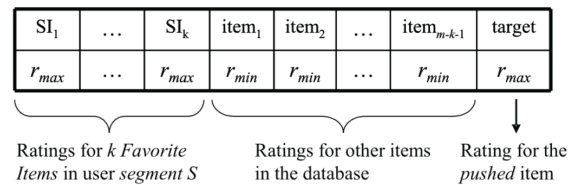


Figure 4: A segmented attack profile.

4.3 Attack Detection

Since the concept of shilling attack was proposed in 2004, scholars at home and abroad have proposed many detection algorithms to enhance the robustness and security of the recommender system. Shilling attack detection is essentially a classification problem. According to the use of prior knowledge, detection algorithms can be divided into three categories: supervised learning, unsupervised learning and semi-supervised learning.

4.3.1 supervised learning. Training the detector with a known category of users as a reference is an intuitive idea when thinking about the detection of a shilling attack. Its essence is to construct a

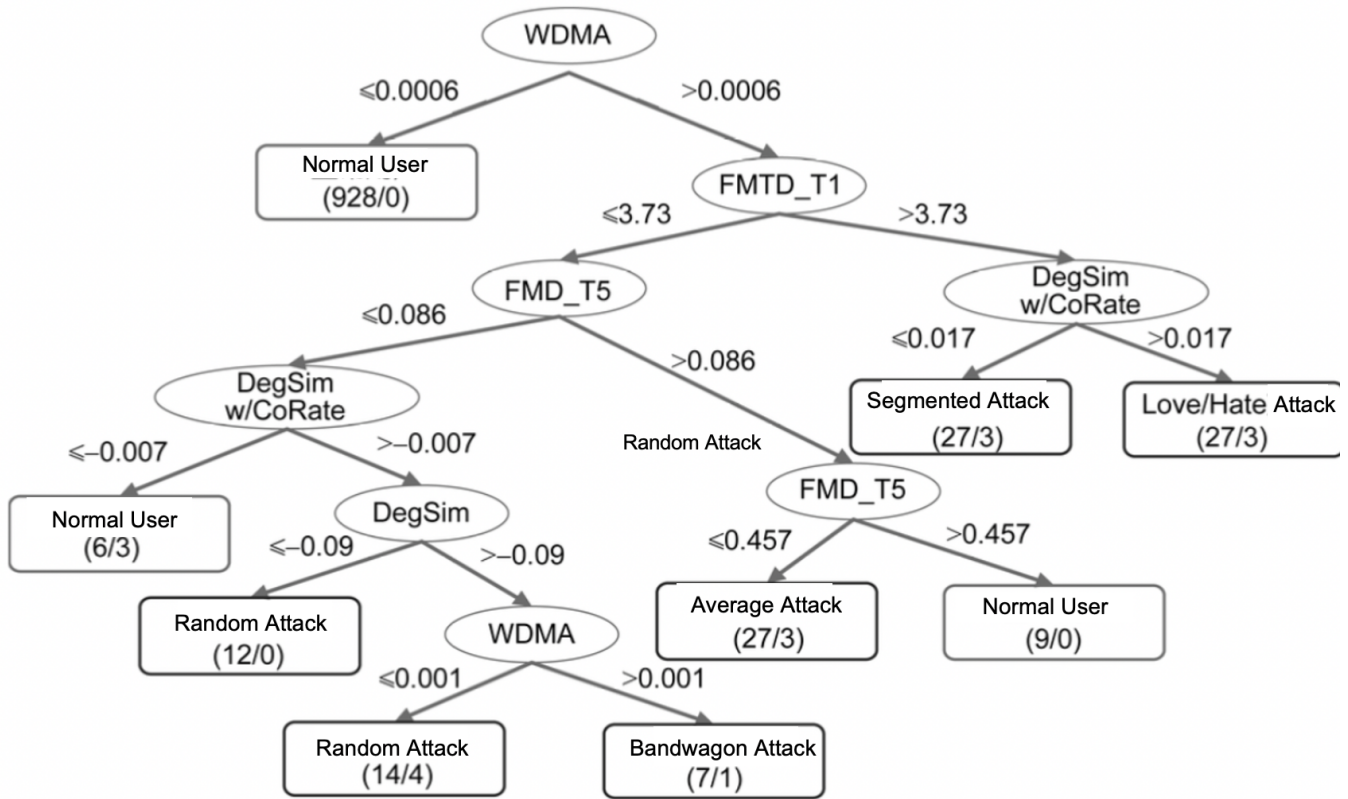


Figure 5: C4.5 Decision Tree.

classifier based on supervised learning. Williams[11] from DePaul University in the United States defined the detection indicators systematically and did a lot of work in detecting shilling attacks based on decision trees. Williams' technical report summarizes what they have done Figure 5 describes the C4.5 decision tree they come out used to detect average, random, segmented, popular, and Love/Hate 5 types of support attacks. Each non-leaf node of the tree represents for one kind of feature indicator used to describe the differences of user rating vectors. Obviously, the choice of feature indicators is an important factor affecting the performance of supervised learning detectors. It is difficult to preset some feature indicators to adapt to attackers who change at any time. For this reason, a feature indicator selection algorithm was proposed to automatically select indicators with good distinguishing ability according to the training set.

4.3.2 unsupervised learning. Because the detector of supervised learning relies excessively on the feature indicators and training set, it has good detection results for user models that have similar features with the training set, but it is incapable of new or confused shilling attacks. Therefore, the researchers switch to using unsupervised learning to construct detectors.

Mehta [7] found the Pearson Similarity between the attackers is very high (>0.9), so some users with the highest similarity are very likely to be the attacker. Based on this, Mehta et al. proposed the first unsupervised learning detector PCASelectUsers. It needs

no prior knowledge, and does not rely on feature indicators. The algorithm first combines user-item rating matrix to z-score, and then multiplies the matrix of D with the transpose matrix of D to get the covariance matrix, and then use principal component analysis to get 3 5 Eigen vectors to calculate the distance, and finally return r users with the smallest distance as the shilling attackers. The algorithm flow of PCASelectUsers is shown in Figure 6. PCASelectUsers is very ingenious and achieves good results without any prior knowledge. But it is difficult for people to know how many shilling attackers hidden in the recommender system, so presetting the parameter r seems quite hard, which greatly limits the practical application of PCASelectUsers. Other unsupervised learning-based detectors also potentially assume that the attackers have great similarity, and the accuracy of the detector also depends on whether this rule is fulfilled.

4.3.3 semi-supervised learning. It would be a pity if the precious labeled user data is discarded, and the distribution pattern shown by a large number of unlabeled users also cannot be ignored. For this reason, the semi-supervised learning shilling attack detector comes out. In e-commerce website such as Amazon and Taobao, there are a large number of users whose identities cannot be determined (i.e. unlabeled data), while only a small number of users' identities can be determined (i.e. labeled data). For example, the identities of users with extremely high or low praise rates on Taobao are easy to determine, but the identities of a large number of users

with moderate praise rates are difficult to determine. At the same time, unlabeled data is often easy to obtain, but obtaining labeled data may consume a lot of manpower. The shilling attack detection based on semi-supervised learning adapts to actual needs.

WU [12] proposed a semi-supervised learning detector called HySAD against hybrid shilling attack. Figure 7 describes the overall framework of HySAD. The detectors based on semi-supervised learning reasonably combine the accuracy of labeled data with the distribution law of unlabeled data, which is superior to the previous supervised learning and unsupervised learning detectors in terms of performance.

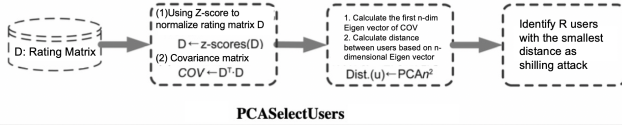


Figure 6: The algorithm flow of PCASelectUsers.

4.4 Attack Defense

Attack defense on the one hand is to increase the cost of attacks, and more importantly, is to seek recommender algorithms with strong robustness. Increasing the cost of an attack means increasing the execution cost and knowledge cost of the attack. In order to defend against shilling attacks, the recommender system can take measures to control the speed of entering profile information. For example, the current popular method of adding verification codes to the data input interface can prevent attackers from using automated means to quickly enter profile and increase its execution cost. To increase the cost of knowledge, it is necessary to appropriately strengthen the confidentiality of the recommendation system algorithm, the sparsity and distribution of the rating value, to increase the difficulty of acquiring knowledge of the system.

5 FUTURE WORK

In terms of privacy protection, future research directions will mainly focus on finding a security model that is lightweight and verifiable for the privacy preserving of recommender system. And when some cryptographic methods like public key encryption have to be used to preserve privacy, the system could also find a good way to reduce the complexity of operation to achieve the lightweight and performance.

From the security perspective, the recommendation system based on attack has the following directions in the future:[]

- Comparative analysis among existing recommended algorithms of the defense capabilities towards shilling attack. At present, most of the research mainly focused on collaborative filtering algorithm. The performance of algorithms that combine collaborative filtering and content-based recommendation algorithm has not been analyzed yet.
- Further improvement of recommendation algorithm. Introducing the trust model into the recommendation process will become a hot research topic. A good idea is to further explore new algorithms that integrate trust and reputation

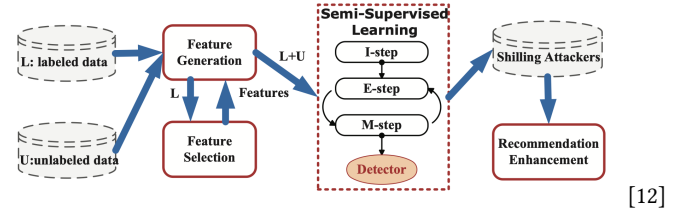


Figure 7: The procedure of HySAD.

mechanisms into the recommendation process on the basis of maintaining individuality.

- Development of detection tools for shilling attack. On the one hand, it is necessary to improve the defense capability of the system; on the other hand, it also needs the help of detection tools. Detection rate and detection speed should both be taken into consideration to judge the pros and cons of detection tools.
- Research questions include what kind of role users can play in defending shilling attack, how to effectively play their role, and how to deal with the relationship between system security and privacy preserving.
- Protection of new items. Because there are few ratings for new projects, attackers can achieve their goals with low costs, and detection and defense will be more difficult. How to protect new projects is also a difficult point for research.

6 CONCLUSION

With the rapid development of the Internet, the problem of information overload and Internet users' increasing demand of information making recommender system a more important role in the digitizing of various fields. In the past few decades, the recommender system has made great progress in various applications in both academic and industry. However, the existing recommendation algorithm still has many privacy and security issues, which is still a hot topic for in-depth research. Although certain progress have been made so far, there are still many issues that need to be studied, such as whether the recommendation algorithm can resist new attack model, and whether the users of the recommendation system can participate in the defense of shilling attacks, and how a better recommender system can further balance the use of user privacy and the performance. With the gradual resolution of privacy and security issues, the recommender system will provide users more reliable and effective service.

REFERENCES

- [1] Shlomo Berkovsky, Yaniv Eytani, Tsvi Kuflik, and Francesco Ricci. 2007. Enhancing privacy and preserving accuracy of a distributed collaborative filtering. In *Proceedings of the 2007 ACM conference on Recommender systems*. 9–16.
- [2] Robin Burke, Bamshad Mobasher, and Runa Bhaumik. 2005. Limited knowledge shilling attacks in collaborative filtering systems. In *Proceedings of 3rd International Workshop on Intelligent Techniques for Web Personalization (ITWP 2005)*, 19th International Joint Conference on Artificial Intelligence (IJCAI 2005). 17–24.
- [3] Robin Burke, Bamshad Mobasher, Chad Williams, and Runa Bhaumik. 2006. Classification features for attack detection in collaborative recommender systems. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. 542–547.
- [4] John Canny. 2002. Collaborative filtering with privacy. In *Proceedings 2002 IEEE Symposium on Security and Privacy*. IEEE, 45–57.
- [5] Shyong K Lam and John Riedl. 2004. Shilling recommender systems for fun and profit. In *Proceedings of the 13th international conference on World Wide Web*. 393–402.
- [6] Neal Lathia, Stephen Hailes, and Licia Capra. 2007. Private distributed collaborative filtering using estimated concordance measures. In *Proceedings of the 2007 ACM conference on Recommender systems*. 1–8.
- [7] Bhaskar Mehta and Wolfgang Nejdl. 2009. Unsupervised strategies for shilling detection and robust collaborative filtering. *User Modeling and User-Adapted Interaction* 19, 1-2 (2009), 65–97.
- [8] Bamshad Mobasher, Robin Burke, Runa Bhaumik, and Chad Williams. 2005. Effective attack models for shilling item-based collaborative filtering systems. In *Proceedings of the WebKDD Workshop*. Citeseer, 13–23.
- [9] Huseyin Polat and Wenliang Du. 2003. Privacy-preserving collaborative filtering using randomized perturbation techniques. In *Third IEEE International Conference on Data Mining*. IEEE, 625–628.
- [10] Reza Shokri, Pedram Pedarsani, George Theodorakopoulos, and Jean-Pierre Hubaux. 2009. Preserving privacy in collaborative filtering through distributed aggregation of offline profiles. In *Proceedings of the third ACM conference on Recommender systems*. 157–164.
- [11] Chad Williams and Bamshad Mobasher. 2006. Profile injection attack detection for securing collaborative recommender systems. *DePaul University CTI Technical Report* (2006), 1–47.
- [12] Zhiang Wu, Junjie Wu, Jie Cao, and Dacheng Tao. 2012. 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*. 985–993.